Oktober '84

Magazin für den Home-Computer-Besitzer

computronic

10/1. Jahrgang

Das Software-Magazin

DM 5,50
öS 45
SFR 5,50

Jetzt noch mehr

aktuelle Informationen

- neue Software-Reviews
- neues vom Markt

Top-Programme wie:

Spiders
Oil-Panic
Fressmann
und
Outlaw
Tic-Tac-Toe
Mother Duck
Cave Man
u. a.

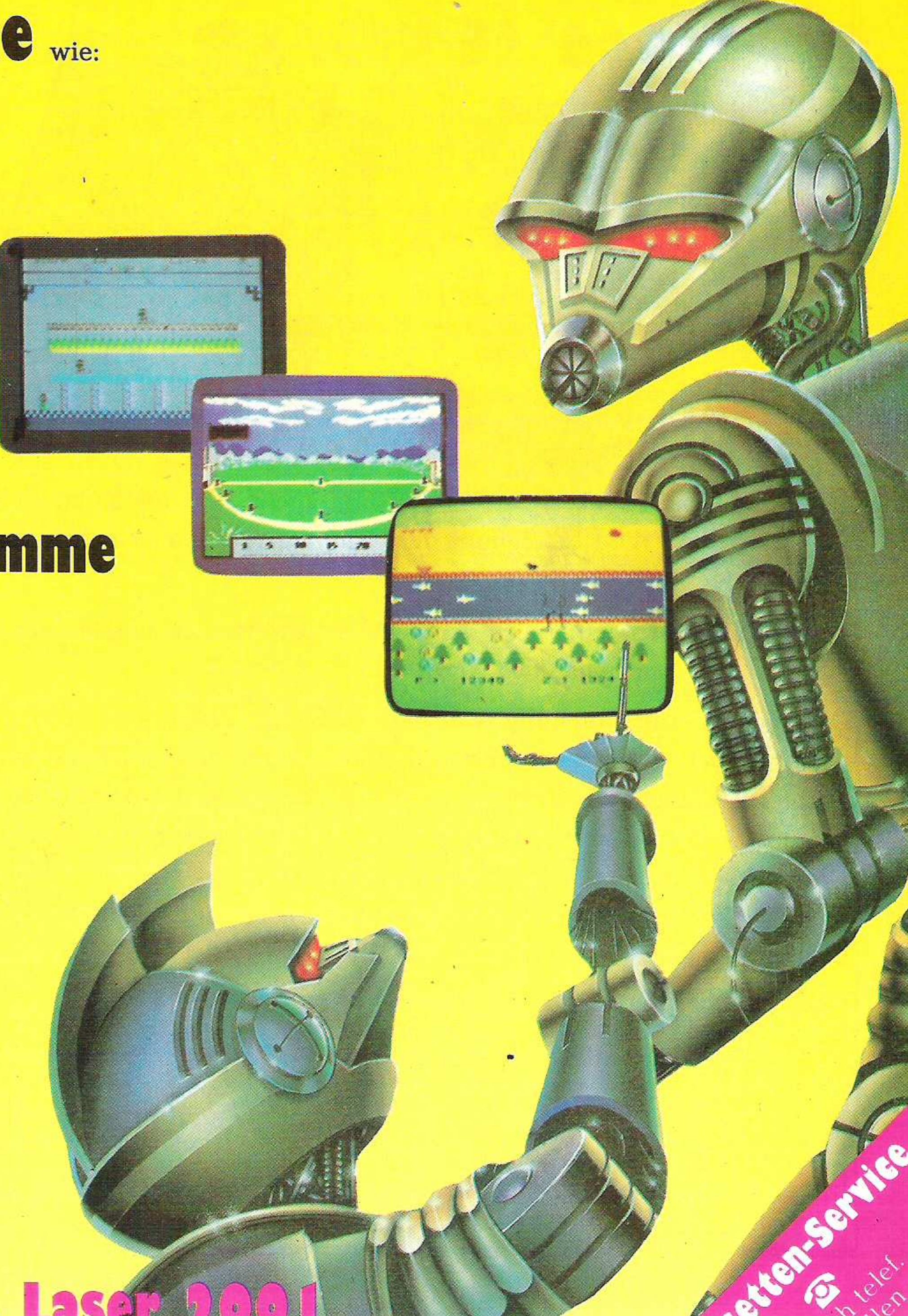
Anwenderprogramme

Screen-Designer
The Basic

Forth-Kurs

verschiedene Zahlensysteme

Commodore 64
TI-99
Atari
VC-20
ZX-81
Apple II
ZX-Spectrum
Dragon 32/64 u. Laser 2001



Kassetten-Service
einfach telef. bestellen

Möchten Sie als Software-Autor für den TRONIC-Verlag tätig werden?

Wir würden uns freuen, in Ihnen einen Ansprechpartner für eine künftige intensive Zusammenarbeit zu finden. Der TRONIC-Verlag vereinbart mit seinen künftigen Software-Autoren pro veröffentlichte volle Seite (einschl. Programmbeschreibung) **ein Honorar von DM 120,–.**

Dieses Entgelt wird fällig, wenn die Redaktion des Verlages sich für eine Veröffentlichung entscheidet. Die Auszahlung erfolgt also nicht erst nach Veröffentlichung in einer unserer Ausgaben, sondern früher.

Der Verlag wird vom Autor berechtigt, seine Manuskripte (Programme) zur Darstellung im Heft heranzuziehen und abzudrucken.

Einzusenden sind:

- Programmbeschreibung
- bespielte Cassette oder Diskette
- Listing (mit Copyright)
- Freiumschlag



Der Autor erklärt sich mit der Lieferung seines Programmes oder seiner Beiträge ausdrücklich bereit, die Verwertung durch den Verlag freizugeben, d. h. er überträgt nicht nur die Nutzung, sondern auch die Übereignung des Computerprogrammes und der Beiträge.

Der Autor verpflichtet sich nur solche Programme anzubieten, die eigene Entwicklungen des Autors sind.

Mit der Veröffentlichung oder dem Anlauf des Programmes und der Beiträge ist es dem Verlag gestattet, auch eine anderweitige bzw. weitergehende Verwertung vorzunehmen, da der Autor dem Verlag das Copyright gegen Honorar gestattet hat. Die Verwertung durch den Verlag ist unbeschränkt und unwiderruflich, wenn nicht 10 Tage nach Zusendung der Unterlagen durch den Autor widersprochen wird.

Haben Sie Interesse? Haben Sie noch Fragen?

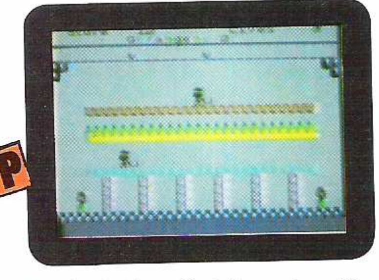
Setzen Sie sich telefonisch mit unserer Redaktion in Verbindung!

TRONIC-VERLAG

DIE REDAKTION

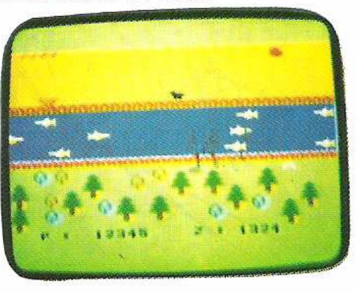
Aus dem Inhalt:

Oil-Panic



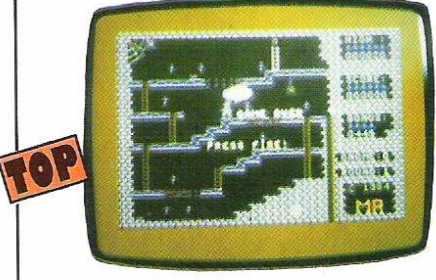
Die beliebte Spielversion für den **ZX-Spectrum 16K** und **48K.**

Mother Duck



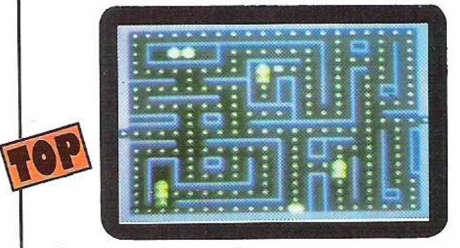
Die Speicherkapazität des **TI-99** wurde wieder voll ausgenutzt.

Spiders



Unser absolutes Topprogramm in dieser Ausgabe, wieder für den **Commodore 64** (siehe auch Heft 9, „Projekt").

Fressmann



Ein Topprogramm für den **VC-20**? Jawohl. Wir finden, trotz bekannter Spielversion, ein für den VC-20 hervorragendes Spiel.

In diesem Heft

10/'84

Software

Aktuelles

LASER 2001 – Ein neuer Stern am Computerhimmel, oder nur eine Sternschnuppe?

Vor einigen Wochen brachte Sanyo Video als Generalimporteur den LASER 2001 auf den Markt. Wir wollten wissen, ob der „große Bruder" der LASER-Familie das hält, was die Werbung verspricht. Sanyo Video stellte uns zu diesem Zweck die komplette LASER 2001-Peripherie zur Verfügung. Unsere Testanlage bestand aus:
LASER 2001-Konsole,
16-K-MEMORY-EXPANSION-MODUL,
DR 10 Datenrecorder,
DI 100 A-DISC-DRIVE-CONTROLLER,
FD 100 A-FLOPPY-DISK-DRIVE,
PC 20 PRINTER CABLE.



Laser 2001

Als Drucker wurde ein Seikosha GP 700 A-Farbdrucker verwendet.
Nach dem Auspacken der Konsole stellten wir mit Freude fest, daß die sonst bei LASER-Computern übliche Gummitastatur (LASER 110 und 210) einer festen Schreibmaschinentastatur gewichen ist.
Neben dem Netzteil sind ein Recorderkabel, ein TV-Anschlußkabel und eine Antennenweiche, die das „umstöpseln" am Fernseher überflüssig macht, beigefügt.
Nachdem die Konsole angeschlossen war, erschien das LASER-LOGO in grüner Schrift auf schwarzem Hintergrund gut lesbar auf dem Bilschirm. Hierzu sei bemerkt, daß der LASER 2001 auf der Unterseite einen kleinen Kanalwahlschalter besitzt, womit der Sendekanal eingestellt werden kann (Kanal 3 oder 4).
Eine rote Kontrollampe signalisiert uns, daß der Rechner in Betrieb ist. Neben dieser Kontrollampe ist eine auffällig große ⟨RESET⟩-Taste zu sehen, auf die wir aber später noch zu sprechen kommen.
Das in ROM gespeicherte Microsoft BASIC entspricht weitestgehend dem Applesoft BASIC, ist jedoch um wesentliche Grafik-, Farb- und Tonbefehle erweitert. Die am meisten gebrauchten Befehle sind direkt über die CONTROL-Tasten erreichbar, können jedoch auch über die Tastatur eingetippt werden.
Der LASER 2001 arbeitet mit einem 6502 A-Mikroprozessor, welcher mit 2 MHz getaktet wird. Somit wird praktisch die doppelte Arbeitsgeschwindigkeit des Apple erreicht.
LASERBASIC kennt 2 verschiedene Bildschirmmodi: 1. Der TEXT-Modus: Hier hat der Rechner eine Auflösung von 24 Zeilen, je 48 Zeichen. Die Zeichen werden mit der PRINT-Anweisung angesprochen und können auch vom BASIC her umdefiniert werden, so daß eine gewisse Grafikfähigkeit auch im TEXT-Modus erreicht werden kann. Im TEXT-Modus ist jedoch nur eine Vordergrund- und eine Hintergrundfarbe möglich.
Die Cursorbewegung läßt sich programmieren, so daß ein Maskenbetrieb möglich ist.

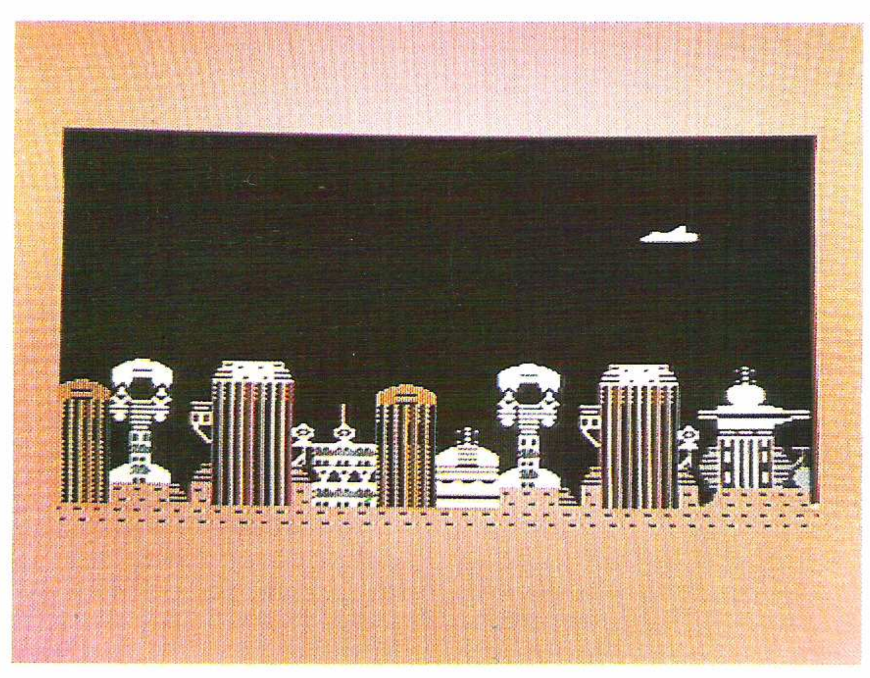
2. Der GRAFIK-Modus:

Im Grafikmodus beträgt die Auflösung 192 Zeilen je 256 Punkten. Hierbei können alle 16 Farben gleichzeitig dargestellt werden. Die Farbauflösung beträgt 192 Zeilen je 32 Farbbalken.
Erst in Grafikmodus kommen die Befehle PLOT, UNPLOT, CIRCLE und RECT zur Wirkung. Die PRINT Anweisung ist im Grafikmodus jedoch wirkungslos, da für den Grafikmodus ein anderer Platz im Videoraum benutzt wird, als für den Textmodus. Es ist deshalb nicht so ohne weiteres möglich, ohne die genauen RAM-Verhältnisse zu kennen, Text und Grafik zu mischen.
Der LASER 2001 benutzt als Videoprozessor den TM 9928 A von Texas Instruments. Dieser Videoprozessor hat die Eigenschaft bis zu 32 Sprites gleichzeitig darstellen zu können. Die Sprites können bis zu einer 32 x 32 Matrix generiert, und ohne Programmkontrolle bewegt werden. LASERBASIC kann jedoch die Sprites nicht direkt ansprechen. Nach Aussage von Sanyo Video ist jedoch ein Charakter- und Sprite-Generator in Vorbereitung, der bis zum Herbst auf Diskette, bzw. Cassette angeboten werden soll. Somit werden auch die Sprites für Benutzer, die sich mit der Maschinensprache nicht auskennen, zugänglich.
All denjenigen Computerfreaks, die sich gerne mit dem Innenleben des Rechners beschäftigen, sei an dieser Stelle jedoch das TM 9928 A Datenbuch von Texas Instruments empfohlen.
Als Soundprozessor wird der SN 76489 A verwendet, welcher 3 Tonkanäle und einen Rauschkanal gleichzeitig verarbeiten kann.
LASERBASIC kennt gleich 2 Routinen, um den Soundprozessor zu steuern:
SOUND: Mit dieser Routine können die drei Tonkanäle und gem. Programmierhandbuch auch der Rauschkanal angesprochen werden. Diese Routine eignet sich besonders gut dazu Melodien zu programmieren. Wir konnten dem LASER 2001 jedoch mittels der SOUND-Routine keinen einzigen Rauschton entlocken, und mußten zu diesem Zweck auf die zweite Routine SGEN zurückgreifen, wodurch sich dieses Vorhaben ausgezeichnet realisieren ließ.
SGEN erlaubt es dem Benutzer ohne umständliches POKEN den Soundprozessor direkt zu steuern. Somit lassen sich alle nur möglichen Geräusche, ob Explosionen, Glockengeläut oder Vogelgezwitscher realisieren. Interessierten Benutzern sei hierbei das SN 76489 A-Datenblatt von Texas Instruments empfohlen.
Kommen wir nun zur Peripherie des LASER 2001:
Als einfacher Massenspeicher läßt sich jeder handelsübliche Kassettenrecorder anschließen. Beim Anschluß eines normalen Recorders muß jedoch darauf geachtet werden, daß entgegen den Angaben der Bedienungsanleitung der rote Stecker an die Ohrhörerbuchse und der schwarze Stecker an die Aufnahmebuchse des Recorders angeschlossen wird. Bei der Verwendung eines handelsüblichen Recorders ergaben sich jedoch durch die unterschiedlichen Recordertypen Schwierigkeiten beim Austausch der Programmkassetten.
Verwendet man hingegen den von Sanyo empfohlenen DR 10 Datenrecorder, so ergeben sich keine Aufnahme- und Ladeschwierigkeiten, da der DR 10 optimal auf den LASER 2001 abgestimmt ist, und eine fest eingestellte Lautstärke besitzt. Die Programmkassetten sind hierbei vollends kompatibel. Auch der Anschluß des Compu

weiter Seite

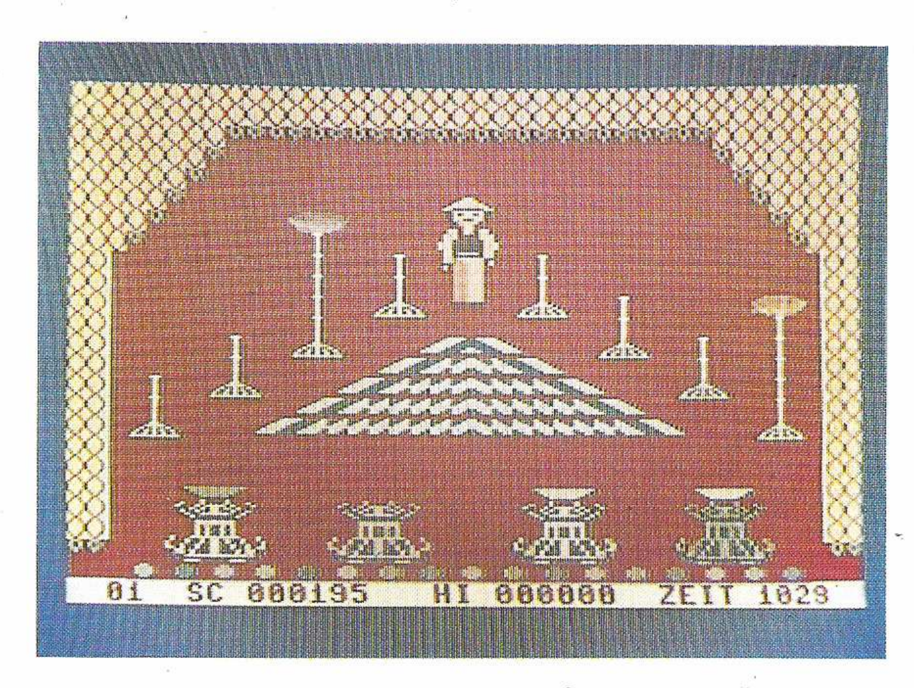
Attraktive Spiele – Software von Dynamics für den Commodore 64

Der große Hamburger Homecomputer-Software- und Zubehör-Anbieter *Dynamics Marketing GmbH* stellt neue, attraktive Spiele-Software vor. Actionreiche, schnelle Spiele, mit guter Grafik, für den Commodore C64. Die Software wird angeboten auf Kassette und teilweise auch auf Diskette. Neu bei den Spielprogrammen auf Kassette ist die Turboladung. Mit diesem Schnellade-Verfahren verkürzt sich die Ladezeit der Kassette auf 2 Minuten.
Gesteuert wird der Turbo-Lader über ein zusätzliches Programm, das auch einen Autostart bietet (der „Run"-Befehl entfällt). Durch den Turbo-Lader ist ein Kassetten-Spielprogramm jetzt genauso schnell auf dem Bildschirm wie bei einer Diskette.



„CAVILON"

Auf dem Planeten Cavilon haben die Computer die Macht übernommen. In allen Gebieten sind Überwachungsbasen gebaut worden, die den Luftraum total kontrollieren. Sie sind mit einem Raumschiff von der Erde mit dem Auftrag unterwegs, die Aufständigen und ihre Raumschiffe zu bekämpfen.
Ein Action-Spiel auf Diskette oder Kassette.



„JONGLEUR"

Auf einer Bühne jongliert ein kleiner Chinese Teller auf langen Stöcken. Damit diese nicht herunterfallen, müssen die Stöcke dauernd in Bewegung gehalten werden. Fällt ein Teller dennoch herunter, muß ein neuer Teller geholt und aufgelegt werden. Sind alle Stöcke belegt, tanzt der kleine Jongleur vor Freude und beginnt die nächste, noch spannendere Runde.
Ein Geschicklichkeitsspiel auf Diskette oder Kassette.



Vielfältiger Drucker für hohe Ansprüche

Der Drucker IDS Modell 480 von Integral Data System, (Paper Tiger) besitzt serienmäßig eine 7/8 Bit Parallel-Schnittstelle (Centronics) bei einer Datentransferrate von 35 Kilobyte und eine RS-232-C-Schnittstelle mit einer wahlweisen DTR von 300, 1200 oder 9600 Baud.
Unter Near Letter Quality werden die Zeichen in einer 24 x 9-Matrix mit 75 Zeichen pro Sekunde dargestellt. Gedruckt werden kann in 10, 12, 16,8 Zeichen pro Inch und Proportionalschrift. Bei 16,8 Zeichen pro Inch bringt der Drucker mit 110 Zeichen in der Sekunde 132 Zeichen in eine Zeile.
Ein selbständiger Randausgleich mit beliebiger Festlegung des linken und rechten Randes weisen diese Maschine als komfortables Gerät aus.
Über die einfache Benutzung durch nur zwei Kontrollcodes ist der Ausdruck auch in Superscript und Subscript möglich.
Im Graphic-Mode erreicht der Drukker eine Auflösung von 84 x 84 Dots per Inch. Einstellbar auf sechs internationale Zeichensätze ist dieser Drukker allen Problemen gewachsen.
Trotz seiner kompakten Maße von nur 115 x 405 x 297 mm (H x B x T) besitzt der IDS 480 einen eingebauten Traktor. Knut Redmann, DV-Anlagen-Großhandel, Postfach 1154, 6073 Egelsbach, Telefon (0 61 03) 4 98 55.

Einzigartiges Disketten-Archivsystem

Albbruck, 17. August 1984. – Disketten sind empfindlich gegen Schmutz. Ein staubdichter Behälter gibt Sicherheit. ERNO hat ein kompaktes Disketten-Aufbewahrungs- und Archivsystem entwickelt, das vor Staub, Rauch, Licht und mechanischen Einflüssen schützt und eine schnelle Ablage, sowie bequemen Zugriff ermöglicht: Die ERNO UNIBOX.
Auf kompaktem Raum wird Platz für 80 Disketten ohne Couvert (bzw. 48 mit Couvert) geboten. 8 Abteile ermöglichen den Aufbau einer übersichtlichen Datei. Die stabilen Trennwände verhindern Beschädigungen, wie Verbiegen und Knicken. Mit ergonomisch gestalteten Griffmulden lassen sich die Abteile leicht mit einer Hand „blättern" und weit öffnen. Die Fächer werden mit dem dazugehörenden Archivierungs-Etiketten-System gekennzeichnet. Das hochwertige Kunststoffmaterial ist „Anti-Statik"-behandelt. Die Gestaltung im modernen Design (Sockelfarbe computerbeige, braun-rauchglasfarbene Abteile) paßt sich jeder Umgebung an. Die UNIBOX beansprucht wenig Raum und findet auch in Schreibtisch-Schüben Platz.
Kompaktbauweise, Maße: L 257 x B 173 x H 143 mm, Gewicht 612 Gramm, empfohlener Verkaufspreis 39,- DM.
Kostenlose Information: ERNO PHOTO GMBH, Postfach 1140, 7892 Albbruck 1, Tel. 0 77 53 / 50 05.
ERNO zählt als Mitglied der international tätigen ERNO Gruppe zu den bekannten Unternehmen der Foto-, Film- und Videobranche. ERNO befaßt sich mit dem Vertrieb namhafter Produkte.

ERNO UNIBOX: Kompaktes Disketten-Aufbewahrungs- und Archivsystem mit direktem Zugriff auf bis zu 80 Disketten im 5¼-Zoll-Format; 8 weit aufklappbare, staubgeschützte Abteile; im aufgeklappten Zustand kein größerer Raumbedarf erforderlich; komplett mit Archivierungs-Etiketten; hochwertiger Kunststoff mit Anti-Statik-Behandlung; Sockelfarbe computerbeige; Abteilfarbe in braun-rauchglas als Lichtschutz; 612 Gramm, Kopaktmaße: L 257 x B 173 x H 143 mm. Preis 39,– DM.

Mikros im Betrieb

Betriebswirtschaftliche Probleme mit dem Mikro-Computer zu lösen, hat der Betriebswirtschafticher Verlag Dr. Th. Gabler, Wiesbaden, mit dem Segment „Mikro" der Programmreihe „Gabler-Praxis" begonnen. Die formulierte Zielsetzung, EDV-Anwenderliteratur herauszugeben, die auch für solche Interessenten lesbar und anwendbar ist, die sich nicht als EDV-Experten verstehen, zeigt sich deutlich bei den ersten drei Titeln, die soeben vorgestellt wurden.
„Statistik im Betrieb mit BASIC auf Commodore", bietet dem Benutzer der Commodore-Serien 2000, 3000, 4000 oder 8000 für den Betrieb mit BASIC-Interpretern Lösungen: Vom Hypothesentest, exponentieller Glättung, Zeitreihenanalysen, Ausgleichsfunktionen bis zu Maß- und Indexzahlen, Regression und Korrelation, Stichprobenauswertung. Diese 45 Programme sind auch auf Diskette erhältlich und lassen sich durch geringfügige Modifikation der Listings problemlos auch an unterschiedliche Systeme anpassen. Die Autoren Scharnbacher und Kastner sind ordentliche Professoren und Dozenten für Statistik an der Fachhochschule in Mainz.
Gustav Kastner ist ebenfalls Autor des zweiten Gabler-Mikro-Buches „Operations Research mit BASIC auf Commodore." Diese besonders zeitintensiven Verfahren werden somit auch dem Klein- oder Mittelbetrieb durch den Mikro zugänglich. Das Buch vermittelt über die Programm-Listings hinaus auch deren Logik anhand von Strukturprogrammen.
Ein für alle Branchen stets aktuelles Problem – die Terminplanung – löst Kastner in seinem nächsten Buch: „Terminplanung mit BASIC auf Commodore". Auch diese Software (ebenfalls 5¼-Zoll-Diskette erhältlich) ist sofort einsetzbar auf alle Commodore-Computern der Serien 2000–4000 mit BASIC-Interpretern.
Gambler sieht dieses neue Programm als Praktiker-Ergänzung der bereits im Markt befindlichen Hobby-Bücher und wird so sehr schnell die Umsetzung wissenschaftlicher Methoden mit Mikro-Computern in Buch- und Diskettenform weiterentwickeln. Für die Qualität der Produkte bürgen neben dem seit Jahren anerkannten Autorenkreis auch Tradition und Erfahrung des Wiesbadener wissenschaftlichen Fachverlages.

Symposium „Mensch + Medien"

mit angeschlossener Ausstellung

Ettlingen/Frankfurt. – Die Präsentation bereits installierter Kunden- und Branchenlösungen mit modernen Kommunikationsmedien soll im Mittelpunkt einer Ausstellung stehen, die vom 19. bis 22. November 1984 in Ettlingen stattfinden wird. Die Ausstellung selbst, bei der vermieden werden soll, eine Ansammlung von Hard- und Software zu präsentieren, ist als Rahmenprogramm für das vom 22. bis 24. November 1984 im Ettlinger Schloß stattfindende Symposium „Mensch + Medien" vorgesehen. Veranstalter ist die Arbeitsgruppe Neue Kommunikationstechniken e.V. (ANK), deren Ziel es ist, neue Medien und die damit auftauchenden Problemfälle, transparenter und – insbesondere der mittelständischen Wirtschaft – zugänglicher zu machen.
Das Symposium selbst, dessen Teilnehmerzahl auf maximal 750 Personen begrenzt ist, teilt sich in Plenarveranstaltungen, Workshops und Anwender-Arbeitsgruppen auf. Referate, wie zum Beispiel „Die neuen Kommunikationstechniken – eine Herausforderung an den Mittelstand" oder „Neue Medien: Evolution oder Revolution in der Gesellschaft?" bilden den Schwerpunkt am 22. November. Die Workshopveranstaltungen am 23. November sind in vier Themenblöcke gegliedert: „Rechnergeschützte Arbeitsplätze", „Mensch + Medien", „Bildschirmtext für Jedermann" und „Anwenderorientierter Beratungs-Service". Abends stehen in der Stadthalle Ettlingen unter dem Thema „Tele-Kommunikation" Informationen über Satelliten-Fernsehen, Kabel-Pilot-Projekte und Video-Konferenzen auf dem Programm. Das Symposium der ANK endet am 24. November 1984 mit dem Treffen verschiedner Anwender-Arbeitsgruppen. Hier werden, so ein Sprecher der Arbeitsgruppe Neue Kommunikationstechniken, teilnehmerorientierte Anwendungserwartungen diskutiert, Daten und Fakten für die Übertragung in die eigene Praxis erstellt, Entscheidungshilfen sowie Richtlinien für Aus- und Weiterbildungs-Konzeption erarbeitet.
Unterlagen über das Symposium *„Mensch + Medien"* sowie die begleitende Ausstellung sind über die ANK-Geschäftsstelle, Ketteler Straße 5, 7517 Waldbronn, anzufordern.

ters stellt keine Schwierigkeiten dar, da die Anschlüsse am DR 10 farblich gekennzeichnet sind. Wer Zeit hat, um Programme und Daten mit einer Geschwindigkeit von 300 Baud (Applesoft-Standard) zu saven, der findet im DR 10 einen sicheren und billigen Massenspeicher.

Der LASER 2001 hat eine eingebaute parallele Schnittstelle. Mittels des PC 20 Printer-Kabels kann somit jeder Drucker mit Centronics-Schnittstelle angeschlossen werden. Man spart sich somit das teure Geld für ein spezielles Interface.

Zur Steuerung der Schnittstelle stehen 2 Befehle zur Verfügung:

LPRINT sendet Daten zum Drucker.
LLIST listet das Programm auf dem Drucker aus.
Der LASER 2001 bietet eine RAM-Kapazität von 32 K-Bytes. Davon stehen in der Grundversion 12884 Bytes für BASIC frei zur Verfügung. Das hört sich zwar recht wenig an, es sei hierzu jedoch bemerkt, daß diese 12884 Bytes volle 7 Seiten Druckerlisting speichern können.
Wem dieser Bereich zu klein ist, der kann das 16 K-Memory-Expansion-Modul in den seitlichen Modulschacht stecken, und somit den BASIC-Bereich um volle 16 K erweitern. Der LASER 2001 bietet dann eine stattliche RAM-ROM-Kapazität von 64 K-Bytes.
Wer auf schnellen Programmen und Datenzugriff Wert legt, der sollte sein Augenmerk auf den DI 100 A-DISK DRIVE-CONTROLLER und das FD 100 A-Floppy-Disk-Drive legen. DI 100 A und FD 100 A werden zusammen ausgeliefert.
Beim Anschluß des Diskettenlaufwerkes ist jedoch unbedingt darauf zu achten, daß die Netzteilstecker der Konsole und des Laufwerkes nicht vertauscht werden, da diese Stecker identisch sind, die Konsole jedoch die Spannung des Laufwerknetzteiles sehr übel nimmt. Die Stecker hätte man hierbei zumindest farblich kennzeichnen sollen.
Im DI 100 A ist das DOS implementiert, welches aus 18 zusätzlichen Befehlen besteht. Das DOS umfaßt sämtliche zur Diskettenverwaltung notwendigen Befehle. Selbst Befehle für RANDOM ACCESS-Dateien sind im DOS enthalten. Somit entfällt die umständliche Programmarbeit, wie sie bei anderen Home Computern üblich ist. Wir vermißten jedoch Befehle, womit sich Dateien und Disketten direkt kopieren lassen.
Ferner mußten wir feststellen, daß das im Handbuch beschriebene „HELLO-Programm" vom DOS nicht ausgeführt werden kann. Um eine Diskette zu initialisieren darf lediglich FORMAT »RETURN« eingegeben werden.

Was uns wiederum gut gefiel ist, daß bei eingelegter Diskette nach dem Einschalten der Konsole automatisch das Inhaltsverzeichnis der Disk gelesen, und auf dem Bildschirm angezeigt wird.

Ferner sind die Befehle zum Dateischutz und Umbenennen einer einzelnen Datei sehr sinnvoll und nützlich. Der Dateibuffer des Controllers nimmt jedoch 1792 Bytes des BASIC-RAMs in Anspruch.

Der DI 100 A-Disk-Drive Controller kann bis zu zwei Laufwerke gleichzeitig verwalten.

Als Manko des Laufwerkes empfanden wir den lauten Datenzugriff, was sich später jedoch als sehr nützlich erwies, da hierdurch niemand auf die Idee kommt, die Diskette beim Datenzugriff trotz eingeschalteter „ACTIVE" Lampe aus dem Laufwerk zu nehmen.

Für Telespieler ist der LASER 2001 das ideale Gerät. Es können 2 handelsübliche Joysticks an der Frontseite der Konsole angeschlossen werden, welche mittels der Routine CALL −1536 auch ins eigene BASIC-Programm eingebunden werden können. Sanyo Video bietet zur Zeit schon eine Reihe von Telespielen auf Modul an. Ferner ist ein Adapter in Vorbereitung, womit Coleco-Telespielkassetten abgespielt werden können. An diesem Coleco-Adapter kann wiederum der von CBS angebotene Atari-Adapter angeschlossen werden. Somit lassen sich auch noch sämtliche Spielmodule für das ATARI VCS 2600-Telespiel abspielen. Dem Spielfan steht damit eine Spielesammlung von mehreren hundert Spielen zur Verfügung.
Auf eine kleine Besonderheit der LASER-Joysticks sei hier hingewiesen: Die LASER-Joysticks besitzen 2 Aktionsknöpfe, welche sich in eigenen BASIC-Programmen unabhängig voneinander abfragen lassen.

Die anfänglich angesprochene "RESET"-Taste ist im Prinzip nur ein Interrupt, d. h. wenn diese Taste gedrückt wird, wird der Computer angehalten, das Programm und sämtliche Daten bleiben jedoch erhalten. Trotzdem sollte die „RESET"-Taste nur im Notfall benutzt werden. Ein BASIC-Programm hält man besser mit BREAK (»CTRL«&»C«) an, da „RESET" das DOS löscht. Sollte dies wirklich einmal passieren, so kann man das DOS jedoch mit CALL 6670 »RETURN« ohne Programmverlust zurückholen.
Die zum LASER 2001 und zum DOS mitgelieferte Literatur läßt zu wünschen übrig. Der Text ist zwar recht ansprechend geschrieben, so daß das Lesen der Bücher Spaß macht, doch sind die Beschreibungen relativ kurz gefaßt, und beschränken sich auf das Notwendigste. In den Programmbüchern sind außerdem einige Fehler zu finden. Sanyo Video überarbeitet und korrigiert jedoch zur Zeit die Handbücher, und wird diese in Kürze auf den Markt bringen.

Dieser Test konnte natürlich nicht alle Seiten des LASER 2001 beleuchten, sondern soll dem Kaufinteressenten einen Einblick in die Leistungsfähigkeit des Computers geben. Ferner sollen durch den Test dem LASER 2001-Benutzer ein paar kleine Tips zur optimaleren Nutzung des Computers gegeben werden. Um den LASER 2001 in allen Details zu beschreiben und zu erforschen könnte man ein ganzes Buch schreiben.

Zusammenfassend kann man sagen, daß der LASER 2001 sein Geld wert ist, und das, was die Werbung verspricht hält. Man bekommt beim LASER 2001 viel Computer fürs Geld. Die kleinen Schwächen, die auftauchten, werden durch andere Stärken vollends wieder wettgemacht, und welcher Computer kann von sich schon behaupten, er wäre „perfekt"? Wir wollen hoffen, daß sich die Literatur bald auch auf den LASER 2001 stürzt, und daß bald eine BASIC-Erweiterung wie zu den kleinen LASER-„Brüdern" erscheinen wird.

Alles in allem kann man sagen, daß der LASER 2001 seine Daseinsberechtigung auf dem Computermarkt verdient hat.

Wir wünschen diesem neuen Stern am Computerhimmel ein langes Leben. Der LASER 2001 ist es wert.

Technische Daten
6502 A-Mikroprozessor 2 MHZ
TM 9928 A-Videoprozessor
SN 76489 A-Sound-Generator
16 K ROM
32 K RAM erweiterbar auf 48 K
Schreibmaschinentastatur mit Groß- und Kleinschreibung
Grafikmodus 192 X 256 Punkte in 16

weiter Seite 73

Splitt



Splitt, der Taucher, soll zu den verborgenen Schätzen tauchen und versuchen, sie zu bergen.

Nach Eingabe von RUN erscheint das Titelbild mit der Bitte „einen Moment bitte". Sobald dieses durch die Aufforderung „press start" ausgetauscht wird, kann das Spiel beginnen. Gespielt wird mit Joystick.

Der Sinn des Spieles besteht darin, den Taucher (rot) durch die Korallen (blau) zu den Schätzen zu bewegen und diese nacheinander zum Boot zu bringen. Es dürfen jedoch nicht zwei Schätze auf einmal aufgenommen werden, da nach Berühren der Wasseroberfläche nur einer registriert wird.
Hat man schließlich alle Schätze geborgen, so gratuliert einem der Computer und das nächste Level beginnt.
Berührt der Taucher während des Spiels eine der blauen Korallen, büßt er unter lautem Getöse eines seiner Leben, die am oberen linken Bildschirmrand angezeigt werden, ein. Zugleich erscheint über ihm eine Sprechblase mit der Inschrift „HELP".

Es kann während des Spieles einmal vorkommen, daß kein Weg durch das Korallenriff existiert, dann bleibt dem Spieler nichts anderes übrig, als eines seiner Bildschirmleben zu opfern um zu einem neuen Bild zu gelangen.

```
10 REM ********************
20 REM * SPLIT DER TAUCHER *
30 REM *     (c)1984 BY     *
40 REM *  OLIVER CYRANKA   *
50 REM ********************
55 GOSUB 950
60 SC=0:LI=4:Z=PEEK(106)-8:H=Z*256
70 REM AENDERN DER ZEICHENSAETZE
80 POKE 53768,26:FOR L=128 TO 471:POKE H
+L,PEEK(57344+I):POKE H+512+L,PEEK(57344
+L):NEXT L
90 READ E:IF E=-1 THEN GOTO 115
100 FOR ABSCH=0 TO 7:READ FIGUR:POKE H+E
*8+ABSCH,FIGUR:NEXT ABSCH
110 GOTO 90
115 GOSUB 1030
120 GRAPHICS 17:SETCOLOR 1,3,9:SETCOLOR
3,1,14:SETCOLOR 2,7,10:POKE 756,Z:SCHATZ
=0:SC=0:SLAB=0:KJ=0
130 DATA 1,8,52,148,120,14,5,2,1
140 DATA 2,16,44,41,30,112,160,64,128
150 DATA 3,0,0,0,40,16,56,108,56
160 DATA 4,0,0,0,16,56,108,56,16
170 DATA 5,0,0,90,189,189,126,60,24
180 DATA 6,0,0,0,0,16,45,255,125
190 DATA 7,0,2,5,136,127,63,31,15
200 DATA 8,25,42,156,108,255,255,255,255
210 DATA 9,0,0,32,124,140,132,148,136
220 DATA 10,255,129,129,255,0,0,0,0
225 DATA 11,129,90,36,90,90,36,90,129
227 DATA 12,255,0,0,0,0,0,0,0
228 DATA 13,0,23,40,68,40,63,162,196
229 DATA 14,32,220,2,64,128,60,64,240
230 DATA 15,0,224,16,44,66,132,2,201
231 DATA 16,65,131,64,32,17,10,4,0
232 DATA 17,0,192,0,8,19,230,8,16
233 DATA 18,21,35,83,141,1,234,20,0
234 DATA 19,24,36,66,64,76,74,36,24
235 DATA 20,0,0,56,68,4,116,68,58
236 DATA 21,0,0,84,42,42,42,42,42
237 DATA 22,0,0,60,66,124,64,66,60
238 DATA 23,0,0,60,66,66,66,66,60
239 DATA 24,0,0,66,66,66,36,24,24
240 DATA 25,0,0,60,66,124,64,66,60
241 DATA 26,0,0,8,84,34,32,32,32
243 DATA -1
245 GOTO 280
250 REM BILDSCHIRMINFORMATIONEN
260 FOR M=0 TO LI-1:COLOR 1:PLOT M,0:NEX
T M:FOR M=10 TO SC+9:COLOR KJ:PLOT M,0:N
EXT M
270 LET KJ=5:RETURN
280 SOUND 3,25,4,1
290 COLOR 10:PLOT 0,2:DRAWTO 19,2
300 COLOR 7:PLOT 9,4:COLOR 8:PLOT 10,4:C
OLOR 9:PLOT 11,4
310 COLOR 12:PLOT 0,5:DRAWTO 19,5
320 COLOR 138:PLOT 0,23:DRAWTO 19,23:PLO
T 2,20:DRAWTO 8,20:PLOT 10,20:DRAWTO 17,
20
```



```
330 PLOT 0,22:DRAWTO 2,20:PLOT 17,20:DRA
WTO 19,22
340 FOR M=0 TO 19:PLOT M,22:NEXT M
350 LET A=35:FOR M=1 TO 16 STEP 2:PLOT M
,22:COLOR A:LET A=A+0.5:NEXT M
360 FOR DU=0 TO 100:COLOR 171:PLOT (RND(
0)*17)+1,(RND(0)*10)+8:NEXT DU
365 PLOT 0,6:DRAWTO 0,21:PLOT 19,6:DRAWT
O 19,21
370 GOSUB 250
380 LET A=9:LET B=6:COLOR 2:PLOT A,B
400 REM BEWEGEN DER OBJEKTE
410 LET S=STICK(0)
420 IF S=7 THEN GOTO 470
430 IF S=11 THEN GOTO 510
440 IF S=14 THEN GOTO 550
450 IF S=13 THEN GOTO 590
460 GOTO 410
470 REM TAUCHEN RECHTS
480 LET A=A+1:LOCATE A,B,H:IF A>18 THEN
LET A=18
490 COLOR 2:PLOT A,B:COLOR 0:PLOT A-1,B
500 GOSUB 630:GOTO 410
510 REM TAUCHER LINKS
520 LET A=A-1:LOCATE A,B,H:IF A<1 THEN L
ET A=1
530 COLOR 1:PLOT A,B:COLOR 0:PLOT A+1,B
540 GOSUB 630:GOTO 410
550 REM TAUCHER AUFWAERTS
560 LET B=B-1:LOCATE A,B,H
570 COLOR 2:PLOT A,B:COLOR 0:PLOT A,B+1
580 GOSUB 630:GOTO 410
590 REM TAUCHER ABWAERTS
600 LET B=B+1:LOCATE A,B,H
610 COLOR 1:PLOT A,B:COLOR 0:PLOT A,B-1
620 GOSUB 630:GOTO 410
630 IF H=12 OR H=171 OR H=138 OR H=35 OR
 H=36 OR H=37 OR H=38 THEN POP :GOTO 650
640 RETURN
650 IF H=35 OR H=36 OR H=37 OR H=38 OR H
=12 THEN GOTO 690
660 COLOR 1:PLOT A,B:COLOR 141:PLOT A-1,
B-2:COLOR 142:PLOT A,B-2:COLOR 143:PLOT
A+1,B-2
670 COLOR 144:PLOT A-1,B-1:COLOR 145:PLO
T A,B-1:COLOR 146:PLOT A+1,B-1
680 LET LI=LI-1:SOUND 0,25,4,15:FOR LAN=
0 TO 100:NEXT LAN:SOUND 0,0,0,0:FOR LAN=
0 TO 900:NEXT LAN
685 IF LI=0 THEN GOTO 750
687 GOTO 120
690 IF H=12 THEN GOTO 720
700 SOUND 0,108,10,15:FOR L=0 TO 100:NEX
T L:SOUND 0,0,0,0
710 LET SCHATZ=1:GOTO 410
720 IF SCHATZ=1 THEN LET SC=SC+1:SOUND 0
,108,10,15:GOSUB 250:FOR LAN=0 TO 100:NE
XT LAN:SOUND 0,0,0,0
730 IF SCHATZ=1 THEN LET SLAB=SLAB+1:IF
SLAB>=7 THEN GOTO 780
740 COLOR 12:PLOT A,B:LET B=B+1:COLOR 2:
PLOT A,B:LET SCHATZ=0:GOTO 410
750 REM GAME OVER
760 LET C=147:FOR ST=5 TO 8:COLOR C:PLOT
 ST,10:LET C=C+1:NEXT ST
770 FOR ST=11 TO 14:COLOR C:PLOT ST,10:L
ET C=C+1:NEXT ST:GOSUB 870:GOSUB 940:LET
 LI=4:GOTO 120
780 REM GESCHAFFT
790 GRAPHICS 18:LET SG=SG+1:SETCOLOR 1,8
,9:POKE 708,58
800 ? #6;"******************"
810 ? #6;"*   gratulation    *"
820 ? #6;"* sie haben den ";SG;". *"
830 ? #6;"*  schatz gehoben  *"
840 ? #6;"*  nun den ";SG+1;".       *"
850 ? #6;"******************"
860 GOSUB 870:GOTO 120
870 REM SOUND
880 RESTORE 910:FOR LESE=0 TO 15:READ N,
LAN
890 SOUND 0,N,10,10:FOR ANHA=0 TO LAN/10
:NEXT ANHA:SOUND 0,0,0,0
900 NEXT LESE
910 DATA 108,500,108,250,144,250,96,250,
85,1000,81,500,85,250,108,250,96,500,0,5
00,81,500,85,250,108,250,72,500
920 DATA 144,500,108,1000
930 RETURN
940 IF PEEK(53279)<>6 THEN GOTO 940
945 RETURN
950 GRAPHICS 18:POKE 708,58:SETCOLOR 1,8
,9:LET RET=RET+1
960 ? #6;"******************"
970 ? #6;"*splitt der taucher*"
980 ? #6;"*     (c)1984 by   *"
990 ? #6;"*  oliver cyranka  *"
1000 ? #6;"*einen moment bitte*"
1010 ? #6;"******************"
1020 IF RET=1 THEN RETURN
1030 POSITION 0,8:? #6;"*   press   start
  *"
1040 IF PEEK(53279)<>6 THEN GOTO 1040
1050 RETURN
```





Das Spiel „PAC MAN“ ist dem gleichnamigen Spielhallenschlager nachempfunden

und läuft auf dem Sinclair ZX Spectrum mit 16k oder 48k RAM.

Da das Programm größtenteils in Maschinensprache geschrieben ist, läuft PAC MAN sehr schnell und erreicht einen großen Schwierigkeitsgrad. Lediglich die Graphik könnte noch verbessert werden (der Spieler ist in den Speicherstellen 32735 bis 32742 dargestellt und die Gegner in 32727 bis 32734).

Das Spiel ist so eingestellt, daß in jedem Bild zweimal das Tempo erhöht wird. Dies kann verändert werden, indem man die BEEP-Befehle in den Zeilen 140 bis 240 verändert. Auch die Spielgeschwindigkeit und die Akustik von PAC MAN kann damit reguliert werden.

Die Punkte werden in den Speicherstellen 31827 und 31828 gescpeichert, und der Highscore in 31824 und 31825, um vor dem RUN-Befehl sicher zu sein.

Mit POKE 23606,8 wird CAPS LOCK so eingestellt, daß jeder Buchstabe als Großbuchstabe eingelesen wird.

Gesteuert wird mit den Tasten Z, X, P, L und Q, um den Spielablauf zu unterbrechen. Nach der Q-Taste kann das Spiel durch Drücken von ›A‹ abgebrochen oder durch eine andere Taste wieder fortgesetzt werden.

Es kann passieren, daß während des Spiels ein Punkt verlorengeht. Daher kann durch Drücken von ›M‹ in Zeile 1 links ein Punkt eingesetzt werden, wenn alle Wege abgefahren sind und dennoch keine neue Phase beginnt.

Bei Beginn einer neuen Runde wird zuerst eine Pause von 4 Sekunden gemacht.

```
  35 REM **  ALEXANDER WALTER
  40 REM **
  45 REM **
  50 REM **
  55 REM **   8900   AUGSBURG
  60 REM **
  65 REM ******************
  70 REM ******************
  80 GO TO 1030
  90
 100 CLS : PRINT CHR$ USR 31002 AND 0
 110 PRINT #1;AT 0,8;"© BY ALEX WALTER";AT 1,1;"8900 AUGSBURG
 120 PRINT USR 31164
 130 IF NOT PEEK 31829 THEN  GO TO 400
 140 IF PEEK 31834 THEN  BEEP .02,0: BEEP .02,6
 150 IF PEEK 31833>128 THEN  BEEP .01,0: BEEP .01,6
 240 BEEP .02,12
 250 IF INKEY$<>"Q" THEN  GO TO 120
 260 LET a$=INKEY$: IF a$<>"" THEN  GO TO 260
 270 LET a$=INKEY$: IF a$="" THEN  GO TO 270
 280 IF a$="A" THEN  GO TO 400
 290 IF a$<>"M" THEN  GO TO 120
 300 POKE 22562,67: GO TO 120
 400 REM    ENDE
 420 LET P=PEEK 31827+PEEK 31828*256
 430 LET H=PEEK 31824+PEEK 31825*256
 440 PRINT AT PEEK 31860,PEEK 31861;"●"
 450 FOR X=1 TO 200: NEXT X
 500 IF H<P THEN  GO TO 600
 540 PRINT #1;AT 0,0;"PUNKTE :",P,"HIGHSCORE :",H;TAB 31;
 550 IF INKEY$<>"" THEN  GO TO 550
 560 PAUSE 0: GO TO 100
 600      POKE 31824,PEEK 31827
 610      POKE 31825,PEEK 31828
 620 PRINT #1;AT 0,0;TAB 10; FLASH 1;"GRATULIERE";
 630 PRINT #1,"   NEUER HIGHSCORE  :  ";P,;
 650 GO TO 550
 999 REM    ANFANG
1000 POKE 23675,87
1010 POKE 23676,127
1020 CLEAR 30977: LOAD ""CODE 30978
1030 PAPER 0: BORDER 0: CLS
1040 INK 7: BRIGHT 1: POKE 23658,8: REM CAPS COCK
1050 PRINT  PAPER 3,,TAB 10;" PAC MAN ",,,
1070 PRINT ''"   BEWEGUNG:"
1080 PRINT '"OBEN    :  P";
1090 PRINT ,"UNTEN   :  L"
1100 PRINT '"LINKS   :  Z";
1110 PRINT ,"RECHTS  :  X"
```

```
1120 PRINT '"PAUSE  :  Q"
1130 PRINT ''"NACH Q KANN MAN"
1140 PRINT '"   DAS SPIEL FORTSETZEN"
1150 PRINT '"MIT  A  DAS SPIEL ABBRECHEN"
1160 PRINT '"MIT  M  DAS SPIEL FORTSETZEN ,","        WOBEI EIN ZUSAETZLICHER","
       PUNKT EINGESETZT WIRD"
1170 PRINT #1;AT 1,0; INK 0; PAPER 4;TAB 7;"DRUECKE EINE TASTE       ";
1180 PAUSE 0
1190 GO TO 100
```

Programmteil 2

```
3000>REM  Eingaberoutine fuer das
3010 REM  Maschienenprogramm  "PACMAN"
3020 REM
3030 REM
3040 REM  Am besten erst dieses Programm
3050 REM  eingeben und dann damit die Werte einpoken.
3070 REM
3080 REM  Nach dem Starten wird die Speicherstelle
3090 REM  und der bisherige Wert angegeben.
3095 REM
3100 REM  Bei einer Eingabe groesser als 256
3110 REM  wird der alte Wert belassen.
3115 REM
3120 REM  Mit einer Zahl kleiner  als 0
3130 REM  geht man um eine Speicherstelle zurueck,
3140 REM  um sie zu korigieren.
3150 REM
3160 REM  Und mit 10000 wird abgespeichert.
3170 REM
3180 REM  Wenn die Controlwerte nicht stimmen
3190 REM  dann hast du einen Tippfehler gemacht !
3200
3210
4000 CLEAR 30977: LET a=30978: REM   *** DAS CLEAR IST WICHTIG ***
4040 LET C1=A: LET C2=A: LET C3=A: LET X=-1
4050 LET X=X+1
4060 LET C=A+X: INPUT (C;"(";PEEK C);") ";Z
4070 LET Z=INT Z: IF Z=10000 THEN  GO TO 5000
4080 IF Z>255 THEN  LET Z=PEEK C
4090 IF Z<0 AND X>0 THEN  LET X=X-1: LET Z=PEEK (A+X): LET C1=C1-Z: LET C2=C2-Z+
2*Z*(C<>2*INT (C/2)): LET C3=C3-Z+2*Z*(C=2*(INT C/2)): GO TO 4060
4100 POKE C,Z: PRINT C;" - ";Z
4200 LET C1=C1+Z: LET C2=C2+Z-2*Z*(C<>2*INT (C/2)): LET C3=C3+Z-2*Z*(C=2*(INT C/
2))
4210 IF X<99 THEN  GO TO 4050
4300 PRINT "SUMME : ";C1'"1. DIVERENZ : ";C2'"2.DIVERENZ : ";C3''
4400 INPUT "ALLES IN ORDNING ? (Y/N) ";A$
4410 IF A$="N" OR A$="n" THEN  GO TO 4060
4420 IF A$<>"Y" AND A$<>"y" THEN  GO TO 4400
4430 LET A=A+100: GO TO 4040
4500
5000 PRINT "*********  WICHTIG  *********"
5010 PRINT "  NOCH ""VOR"" DEM NEULADEN "
5020 PRINT "  ***  CLEAR 30977 *** EINGEBEN"
5030 PRINT "SONST LERNST DU UNTER UMSTAENDEN DAS FLUCHEN"
5050 SAVE "PAC CODE" LINE 4060
5060 SAVE "PAC CODE"CODE 30978,1790
```

```
30978  237  111  178  186  32   14   62   67   50   143
30988  92   122  215  62   4    50   143  92   122  201
30998  245  215  241  201  46   0    14   1    62   8
31008  205  177  121  33   159  124  1    96   1    22
31018  144  62   4    50   143  92   175  205  2    121
31028  205  2    121  11   237  111  35   120  177  32
31038  241  62   8    14   15   46   10   205  177  121
31048  62   70   50   143  92   58   85   124  167  32
31058  2    62   4    71   5    40   5    62   161  215
31068  24   248  62   71   50   116  89   50   107  89
SUMME : 40497       1. DIVERENZ : 31237        2.DIVERENZ : 21459

31078  50   129  90   50   158  90   50   33   88   50
31088  62   88   33   121  124  17   83   124  1    38
31098  0    237  176  201  221  126  2    133  254  255
31108  32   2    62   21   254  22   32   1    175  111
31118  221  126  3    129  230  31   79   229  38   0
31128  41   41   41   41   41   6    88   9    235  225
31138  26   254  4    201  254  66   200  254  69   200
31148  50   143  92   62   144  245  62   22   215  125
31158  215  121  215  241  215  201  221  33   91   124
31168  6    4    197  221  110  0    221  78   1    221
SUMME : 42052       1. DIVERENZ : 31048        2.DIVERENZ : 19904

31178  126  4    205  166  121  175  50   86   124  205
31188  191  123  205  126  121  32   86   221  110  0
31198  221  78   1    58   86   124  167  32   56   60
31208  50   86   124  221  126  2    167  32   12   205
31218  9    122  221  119  2    175  221  119  3    24
31228  217  205  9    122  221  119  3    175  221  119
31238  2    24   205  197  58   120  92   79   58   82
31248  124  230  31   71   10   169  50   82   124  230
31258  1    193  192  61   201  221  126  2    237  68
31268  221  119  2    221  126  3    237  68   221  119
SUMME : 42466       1. DIVERENZ : 31078        2.DIVERENZ : 19490
31278  3    24   165  205  191  123  221  119  4    58
31288  87   124  50   143  92   62   160  205  177  121
31298  221  117  0    221  113  1    17   5    0    193
31308  221  25   5    194  194  121  221  33   111  124
31318  221  110  0    221  78   1    245  205  191  123
31328  1    0    0    122  148  48   3    5    24   3
31338  40   1    4    123  149  48   3    13   24   3
31348  40   1    12   221  126  2    237  68   185  32
31358  3    221  119  2    221  126  3    237  68   184
31368  32   3    221  119  3    58   87   124  103  62
SUMME : 40526       1. DIVERENZ : 31118        2.DIVERENZ : 21430

31378  160  205  89   123  205  191  123  241  221  33
31388  116  124  1    254  254  175  237  104  203  77
31398  32   1    61   203  85   32   1    60   87   175
31408  6    223  237  104  203  69   32   1    61   6
31418  191  237  104  203  77   32   1    60   66   79
31428  62   161  221  54   4    0    38   70   205  89
31438  123  221  126  4    221  54   4    0    205  6
31448  123  58   88   124  167  40   20   61   50   88
31458  124  6    69   254  10   48   6    230  2    32
31468  2    6    66   120  50   87   124  62   71   50
SUMME : 40949       1. DIVERENZ : 31075        2.DIVERENZ : 21007
```

31478	143	92	14	15	46	12	62	8	205	177
31488	121	237	75	83	124	201	254	67	40	18
31498	254	71	192	42	83	124	17	30	0	25
31508	34	83	124	62	30	50	88	124	42	83
31518	124	17	5	0	25	34	83	124	42	89
31528	124	43	34	89	124	124	181	192	42	83
31538	124	229	58	85	124	245	205	26	121	241
31548	225	50	85	124	6	200	118	16	253	14
31558	100	9	34	83	124	33	43	45	209	235
31568	167	237	82	235	32	248	229	24	154	245

SUMME : 41226 1. DIVERENZ : 31180 2.DIVERENZ : 20730

31578	229	197	221	110	0	221	78	1	221	126
31588	4	205	166	121	217	221	70	3	221	78
31598	2	209	122	179	40	31	122	131	203	71
31608	32	10	120	167	32	1	95	121	167	32
31618	1	87	221	115	2	221	114	3	217	205
31628	126	121	217	40	2	75	66	221	112	3
31638	221	113	2	217	221	110	0	221	78	1
31648	205	126	121	32	9	221	110	0	221	78
31658	1	221	126	4	221	117	0	221	113	1
31668	221	119	4	241	50	143	92	241	195	177

SUMME : 43059 1. DIVERENZ : 31199 2.DIVERENZ : 19897

31678	121	237	91	116	124	97	229	167	237	82
31688	225	192	58	88	124	167	32	68	193	193
31698	58	85	124	61	50	85	124	202	243	122
31708	198	13	79	46	10	205	149	121	175	18
31718	221	33	91	124	6	6	197	221	110	0
31728	221	78	1	221	126	4	205	166	121	193
31738	17	5	0	221	25	16	235	17	91	124
31748	33	129	124	1	30	0	237	176	6	40
31758	118	16	253	195	243	122	42	83	124	17
31768	100	0	25	34	83	124	221	126	4	205

SUMME : 42494 1. DIVERENZ : 32370 2.DIVERENZ : 20462

31778	6	123	221	54	0	1	221	54	1	1
31788	42	116	124	124	214	11	56	5	125	214
31798	15	48	8	221	54	0	20	221	54	1
31808	30	221	110	0	221	78	1	205	149	121
31818	221	119	4	195	75	123	0	0	0	0
31828	0	0	0	0	0	0	0	0	0	0
31838	0	0	0	0	0	0	0	0	0	0
31848	0	0	0	0	0	0	0	0	0	0
31858	0	0	0	0	0	0	0	0	0	4
31868	0	66	0	84	1	2	4	1	0	0

SUMME : 35868 1. DIVERENZ : 31042 2.DIVERENZ : 27088

31878	1	27	1	0	0	17	5	1	0	0
31888	15	26	1	0	0	5	13	1	0	0
31898	19	15	255	0	0	89	153	153	153	217
31908	128	25	153	153	156	1	157	153	153	153
31918	156	96	0	0	0	96	0	0	0	0
31928	6	0	6	0	0	0	6	96	89	153
31938	192	96	64	64	24	4	6	4	6	5
31948	152	4	6	96	96	0	96	32	96	96
31958	0	6	6	6	2	2	0	6	6	96
31968	96	64	96	0	96	96	64	26	2	6

SUMME : 36396 1. DIVERENZ : 32024 2.DIVERENZ : 26560

```
31978   0     0     4     6     6     96    32    96    32    25
31988   160   32    96    0     0     3     153   153   154   2
31998   6     96    0     96    0     0     0     0     57    153
32008   156   0     0     0     0     0     6     96    25    189
32018   128   64    89    128   0     0     3     156   4     1
32028   217   152   6     96    0     6     0     96    96    0
32038   89    156   0     6     6     0     96    0     6     57
32048   152   2     1     160   96    89    175   243   156   6
32058   3     128   96    25    154   0     0     0     0     0
32068   32    111   255   255   246   2     0     0     96    0
SUMME : 37549     1. DIVERENZ : 31585         2.DIVERENZ : 25407

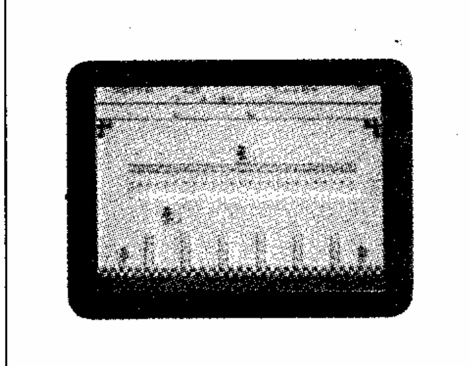
32078   0     89    153   152   1     192   0     111   255   255
32088   246   0     5     128   57    156   4     96    0     0
32098   0     57    128   111   255   255   247   153   154   0
32108   0     6     6     96    89    128   64    0     0     57
32118   217   153   154   0     0     4     4     6     6     96
32128   96    0     96    89    128   0     96    0     0     1
32138   156   6     6     6     6     96    32    64    96    96
32148   0     64    96    25    152   0     6     6     2     2
32158   6     96    0     96    96    96    64    96    32    0
32168   0     4     6     6     0     0     6     121   153   160
SUMME : 38785     1. DIVERENZ : 31923         2.DIVERENZ : 25171

32178   96    96    32    32    0     88    4     2     6     6
32188   1     153   158   96    0     0     96    96    0     0
32198   64    96    6     0     6     6     0     0     6     96
32208   25    128   32    32    25    128   32    96    27    152
32218   2     2     1     152   6     96    0     0     0     0
32228   0     0     0     96    0     0     0     0     0     0
32238   6     57    153   153   153   153   128   25    153   185
32248   152   1     153   153   153   153   154   0     33    15
32258   91    126   35    34    0     91    111   60    200   38
32268   0     41    41    41    237   75    54    92    9     62
SUMME : 37823     1. DIVERENZ : 31415         2.DIVERENZ : 26133
```

Oil-Panic

Topprogramm

Die beliebte Spielversion, die defekte Ölleitung, jetzt auch für den ZX-Spectrum 48 K.



Oil-Panic läuft jedoch auch auf der 16 K-Erweiterung, wenn die Anfangsmelodie, also Zeile 20–55, nicht eingegeben wird.

In der Spielversion erscheint eine Figur, welche vom Spieler gelenkt wird. Dieser muß versuchen, möglichst viele Tropfen aufzufangen, die aus einer undichten Ölleitung heraustreten. Aufgepaßt – denn der Eimer, der zum auffangen der Tropfen da ist, ist mit nur vier Tropfen gefüllt. Wieviel Tropfen im Eimer sind, wird hinter „GEFAN-



GEN" angezeigt. Ist genügend Öl aufgefangen worden, muß es in den Behälter des unteren Männchens gegossen werden. Diese Figur bleibt jedoch nicht unbeweglich stehen, sondern erschwert das Umgießen durch ständiges Herumlaufen. Für das Auffangen eines Tropfens bekommen Sie 2 Punkte. Für einen vollen, umgegossenen Eimer 10 Punkte. Wird der volle Eimer daneben geschüttet, fällt die Spielfigur an der jeweiligen Seite herunter. Läuft der Eimer über, oder wird das kostbare Öl vergossen, wird ein Spielversuch abgezogen.

```
   0 >REM ©©©©©©©©©©©©©©©©©©©©©©
          ©      Oil-Panic      ©
          ©         by          ©
          ©   Oliver Kurth      ©
          ©                     ©
          ©©©©©©©©©©©©©©©©©©©©©©
   5:
   6:
   7:
  10 CLS : BORDER 1: POKE 23756,
0
  20 BEEP 1,0: BEEP 1,2: BEEP .5
,3: BEEP .5,2: BEEP 1,0
  30 BEEP 1,0: BEEP 1,2: BEEP .5
,3: BEEP .5,2: BEEP 1,0
  35 BEEP 1,3: BEEP 1,5: BEEP 2,
7
  40 BEEP 1,3: BEEP 1,5: BEEP 2,
7
  45 BEEP .75,7: BEEP .25,8: BEE
P .5,7: BEEP .5,5: BEEP .5,3: BE
EP .5,2: BEEP 1,0
  50 BEEP .75,7: BEEP .25,8: BEE
P .5,7: BEEP .5,5: BEEP .5,3: BE
EP .5,2: BEEP 1,0
  55 BEEP 1,0: BEEP 1,-5: BEEP 2
,0: BEEP 1,0: BEEP 1,-5: BEEP 2,
0
  95 CLS
 100 FOR a=144 TO 164: FOR b=0 T
O 7: READ c: POKE USR CHR$ a+b,c
: NEXT b: NEXT a
 101 DATA 58,41,146,108,108,146,
41,68
 102 DATA 255,137,147,165,201,14
5,161,193
 103 DATA 255,255,0,32,0,0,0,0
 104 DATA 0,0,0,0,0,0,210,255
 105 DATA 64,97,50,146,117,24,36
,255
 106 DATA 0,126,165,145,165,126,
24,24
 107 DATA 108,188,188,186,168,40
,68,231
 108 DATA 0,0,0,0,0,60,60,255
 109 DATA 255,195,60,195,60,195,
60,255
 110 DATA 0,0,129,129,129,129,12
6,0
 111 DATA 16,56,56,56,16,255,169
,255
 112 DATA 56,70,101,145,153,165,
102,24
 113 DATA 4,9,146,85,74,160,120,
120
 114 DATA 0,30,1,7,127,222,192,6
4
 115 DATA 0,120,192,253,135,123,
3,3
 116 DATA 9,16,46,120,252,228,10
4,48
 117 DATA 39,100,184,80,81,165,1
22,44
 118 DATA 56,124,255,0,124,254,2
44,223
 119 DATA 232,56,19,124,184,40,7
2,102
 120 DATA 25,62,255,0,62,127,47,
251
 121 DATA 23,25,200,62,30,21,18,
102
 300 CLS : BORDER RND*4: PRINT A
T 0,2;"Score :";AT 0,20;"Lives :
";AT 1,9;"Gefangen:"
 310 FOR a=0 TO 31: BEEP .01,40:
 PRINT AT 2,a;"ö";AT 3,a;"ö": NE
XT a: PRINT AT 4,0;"▟▟";AT 5,0;"
▟";AT 4,30;"▙▙";AT 5,31;"▙"
 320 PRINT AT 3,9;"Ü";AT 3,17;"Ü
"
 330 FOR b=4 TO 28: BEEP .008,20
: BEEP .01,30: PRINT AT 9,b;"I";
AT 11,b;"K";AT 12,b;"▙";AT 16,b;
"A": NEXT b
 340 FOR c=6 TO 26 STEP 4: BEEP
c;"ä";AT 19,c;"ä";AT 20,c;"ä": N
EXT c
 350 PRINT AT 18,3;"§";AT 18,29;
"§";AT 19,3;"Ü";AT 19,29;"Ü";AT
20,3;"ß";AT 20,29;"ß": FOR t=0 T
O 31: BEEP .01,55: PRINT AT 21,t
;"▚": NEXT t
 360 PRINT AT 8,16;"SJ";AT 7,16;
"R"
 400 LET DF=VAL "1": LET AT=VAL
"9": LET V=VAL "9": LET A=VAL "1
6": LET B=VAL "8": LET C=VAL "15
": LET D=VAL "16": LET X=VAL "0"
: LET LIV=VAL "10": LET SCO=VAL
"0": LET GEF=VAL "0"
 498 LET F=VAL "4"
 499 LET E=INT (RND*15)
 500 IF INKEY$="8" THEN LET A=A+
2: BEEP .005,50: PRINT AT 6,A;"S
";AT 7,A;"R";AT 8,A+1;"J";AT 6,A
-2;"  ";AT 7,A-2;"  ";AT 8,A-3;"
"
 501 IF SCO>VAL "80" AND DF=VAL
"1" THEN PRINT AT 13,11;"Bonusru
nde": FOR Y=0 TO 6: BEEP .04,35:
 BEEP .07,40: BEEP .1,45: BEEP .
5,50: NEXT Y: LET LIV=LIV+VAL "3
": PRINT AT 13,11;"           ":
LET DF=VAL "0": GO TO VAL "500"
 502 IF GEF>VAL "4" THEN LET GEF
=VAL "0": LET LIV=LIV-VAL "1": L
ET SCO=SCO-VAL "2"
 503 PRINT AT 0,10;SCO;AT 1,19;G
EF;AT 0,29;"  ";AT 0,29;LIV
 504 PRINT AT 9,9;"I";AT 9,17;"I
"
 505 IF INKEY$="5" THEN LET A=A-
2: BEEP .005,46: PRINT AT 6,A;"U
";AT 7,A;"T";AT 8,A-1;"J";AT 6,A
+1;"  ";AT 7,A+2;"  "
 506 LET G=INT (RND*2)
 507 BEEP .001,40
 508 IF X=1 AND G=1 THEN GO TO 9
00
 509 IF LIV=VAL "0" THEN GO TO V
AL "3000"
 510 IF X=0 AND G=1 THEN LET D=D
+2: BEEP .005,20: PRINT AT 14,D;
"R";AT C,D;"SJ";AT C,D-3;"   ";A
T 14,D-2;"  ": IF D>27 THEN LET X
=1
 511 LET AT=VAL "9"
 515 IF INKEY$="9" AND A=VAL "28
" THEN GO TO VAL "3100"
 516 IF INKEY$="4" AND A=VAL "4"
THEN GO TO VAL "3200"
 520 IF A>VAL "28" THEN GO TO VA
L "3500"
 521 IF A<VAL "4" THEN GO TO VAL
"3600"
 600 IF E=VAL "1" THEN GO TO VAL
"920"
 601 IF E=VAL "2" THEN GO TO VAL
"950"
 603 IF E>VAL "2" OR E<VAL "1" T
HEN GO TO VAL "498"
```

```
 800 GO TO VAL "500"
 900 LET D=D-VAL "2": BEEP .005,30: PRINT AT 14,D;"T";AT C,D-1;"JU";AT C,D+1;"  ";AT 14,D+2;" ": IF D<5 THEN LET X=0
 905 GO TO VAL "500"
 920 LET F=F+VAL "1": BEEP .01,20: PRINT AT F,9;"P": IF F>4 THEN PRINT AT F-1,9;" "
 921 IF F<VAL "8" THEN GO TO 500
 923 IF A+VAL "1"=VAL "9" OR A-1=VAL "9" AND F=VAL "8" THEN LET SCO=SCO+VAL "2": LET GEF=GEF+VAL "1": PRINT AT 8,9;"J": GO TO VAL "498"
 925 IF F=VAL "9" THEN PRINT AT F,9;"I": LET LIV=LIV-VAL "1": GO TO VAL "498"
 930 GO TO VAL "500"
 950 LET F=F+VAL "1": BEEP .01,48: PRINT AT F,17;"P": IF F>4 THEN PRINT AT F-1,17;" "

";AT 7,A;"T";AT 8,A-1;"J";AT 8,A+1;"  ";AT 7,A+2;" "
 506 LET G=INT (RND*2)
 507 BEEP .001,40
 508 IF X=1 AND G=1 THEN GO TO 900

 509 IF LIV=VAL "0" THEN GO TO VAL "3000"
 510 IF X=0 AND G=1 THEN LET D=D+2: BEEP .005,20: PRINT AT 14,D;"R";AT C,D;"SJ";AT C,D-3;"  ";AT 14,D-2;" ": IF D>27 THEN LET X=1
 511 LET AT=VAL "9"
 515 IF INKEY$="8" AND A=VAL "28" THEN GO TO VAL "3100"
 516 IF INKEY$="4" AND A=VAL "4" THEN GO TO VAL "3200"
 520 IF A>VAL "28" THEN GO TO VAL "3500"
 521 IF A<VAL "4" THEN GO TO VAL "3600"
 600 IF E=VAL "1" THEN GO TO VAL "920"
 601 IF E=VAL "2" THEN GO TO VAL "950"
 603 IF E>VAL "2" OR E<VAL "1" THEN GO TO VAL "498"
```



```
 800 GO TO VAL "500"
 900 LET D=D-VAL "2": BEEP .005,30: PRINT AT 14,D;"T";AT C,D-1;"JU";AT C,D+1;"  ";AT 14,D+2;" ": IF D<5 THEN LET X=0
 905 GO TO VAL "500"
 920 LET F=F+VAL "1": BEEP .01,20: PRINT AT F,9;"P": IF F>4 THEN PRINT AT F-1,9;" "
 921 IF F<VAL "8" THEN GO TO 500
 923 IF A+VAL "1"=VAL "9" OR A-1=VAL "9" AND F=VAL "8" THEN LET SCO=SCO+VAL "2": LET GEF=GEF+VAL "1": PRINT AT 8,9;"J": GO TO VAL "498"
 925 IF F=VAL "9" THEN PRINT AT F,9;"I": LET LIV=LIV-VAL "1": GO TO VAL "498"
 930 GO TO VAL "500"
 950 LET F=F+VAL "1": BEEP .01,48: PRINT AT F,17;"P": IF F>4 THEN PRINT AT F-1,17;" "
```



```
 953 IF F<8 THEN GO TO 500
 955 IF A+1=VAL "17" OR A-1=VAL "17" AND F=VAL "8" THEN LET SCO=SCO+VAL "2": LET GEF=GEF+VAL "1": PRINT AT 8,17;"J": GO TO VAL "498"
 960 IF F=VAL "9" THEN PRINT AT F,17;"I": LET LIV=LIV-VAL "1": GO TO VAL "498"
 970 GO TO VAL "500"
3000 FOR S=0 TO 3: FOR U=0 TO 5: BEEP .05,30: BEEP .5,40: BORDER RND*6: NEXT S: NEXT U
3010 FOR K=0 TO 20: BEEP .008,K: BEEP .01,K: BORDER RND*6: NEXT K
3020 PRINT AT 13,11;"████████████";AT 14,10;"█Game*Over█";AT 15,11;"████████████"
3030 BEEP 1,10: BEEP 1,8: BEEP 1,7: BEEP .5,5: BEEP .5,6: BEEP .5,6: BEEP .1,4: BEEP .1,4: BEEP

.09,1: BEEP .09,1: BEEP .06,2: BEEP .06,2: BEEP .06,2: BEEP .04,0: BEEP .04,0: BEEP .04,0: BEEP .01,-2: BEEP .01,-2: BEEP .01,-2
3035 CLS : PRINT AT 11,9;"NOCHEINMAL?";AT 8,5;"Punkte erreicht:";SCO: PAUSE 4E4
3040 IF INKEY$="j" THEN GO TO 30
3050 STOP
3100 PRINT AT 8,29;"O"
3105 LET AT=AT+1: BEEP .02,29: PRINT AT AT,29;"P": IF A>9 THEN PRINT AT AT-1,29;" "
3106 IF AT=VAL "15" AND D=VAL "28" AND GEF<VAL "3" THEN LET LIV=LIV-VAL "1": GO TO VAL "500"
3109 IF AT=VAL "15" AND D=VAL "28" AND GEF>VAL "2" THEN LET SCO=SCO+10: PRINT AT 15,29;"J": LET GEF=VAL "0": FOR h=1 TO 5: BEEP .01,10: BEEP .06,20: BEEP .1,30:

 NEXT h: GO TO VAL "500"
3110 IF AT=VAL "17" THEN PRINT AT AT,29;"M": BEEP .01,60: BEEP .1,60: BEEP .01,60: LET LIV=LIV-VAL "1": LET GEF=VAL "0": PRINT AT 0,29;LIV;AT 1,19;GEF;AT 17,29;" ": GO TO VAL "500"
3115 GO TO VAL "3105"
3200 PRINT AT 8,3;"N"
3201 LET AT=AT+VAL "1": BEEP .02,20: PRINT AT AT,3;"P": IF AT>9 THEN PRINT AT AT-1,3;" "
3203 IF AT=VAL "15" AND D=VAL "4" AND GEF<VAL "3" THEN LET LIV=LIV-VAL "1": GO TO VAL "500"
3205 IF AT=VAL "15" AND D=VAL "4" AND GEF>VAL "2" THEN LET SCO=SCO+VAL "10": PRINT AT 15,3;"J": LET GEF=VAL "0": FOR t=1 TO 6: BEEP .01,2: BEEP .06,12: BEEP .1,22: NEXT t: GO TO VAL "500"
3210 IF AT=VAL "17" THEN PRINT A

T AT,3;"M": BEEP .01,60: BEEP .1,60: BEEP .1,60: LET LIV=LIV-VAL "1": LET GEF=VAL "0": PRINT AT 17,3;" ";AT 1,19;GEF;AT 0,29;LIV: GO TO VAL "500"
3215 GO TO VAL "3201"
3500 FOR O=9 TO 17: BEEP .01,O: PRINT AT O,29;"L";AT O-1,29;"0";AT O-2,29;" ": NEXT O: BEEP .01,55: BEEP .01,56: BEEP .01,57: BEEP .01,58: BEEP .01,59: BEEP .01,60: PRINT AT 16,29;" ";AT 17,29;" ": LET LIV=LIV-VAL "1": LET GEF=VAL "0": PRINT AT 0,29;LIV;AT 1,19;GEF
3502 GO TO VAL "500"
3600 PRINT AT 8,0;"    ";AT 7,0;"    ": FOR V=9 TO 17: BEEP .01,V: PRINT AT V,3;"L";AT V-1,3;"0";AT V-2,3;" ": NEXT V: BEEP .01,55: BEEP .01,56: BEEP .01,57: BEEP .01,58: BEEP .01,59: BEEP .01,60: PRINT AT 16,3;" ";AT 17,3;" ": LET LIV=LIV-VAL "1": LET GEF=VAL "0": PRINT AT 0,29;LIV;AT 1,19;GEF
3601 GO TO VAL "500"
```

Panik-Labyrinth

Panik-Labyrinth ist in erster Linie ein Strategie- und Übersichtspiel, das Können und Kombinationsgabe vom Spieler erfordert.

Sie befinden sich in einem Labyrinth und werden von einem Geist gejagt. Jede Sekunde, die Sie überleben, bringt einen Punkt, aber, im Labyrinth liegen auch Schätze in Form von Gold-Dollars, die, wenn Sie welche einsammeln, jede Menge Punkte bringen.

Keine leichte Aufgabe, denn der Geist weiß immer, wo Sie sich befinden und wird Sie jagen, bis er Sie erwischt hat. Da er nunmal ein Geist ist, kann er selbstverständlich auch durch Mauern gehen, ohne dabei ein Loch zu hinterlassen. Sie sind an die Wege des Labyrinths gebunden.

Damit Sie überhaupt dem Geist entwischen können, haben Sie eine Gabe, von der andere Menschen nur träumen. Sie können sich im Nu in Luft auflösen und an einer anderen Stelle wieder erscheinen. Dies haben Sie einem kleinen Gerät zu verdanken, einem Transmitter: Allerdings benötigen Sie, um zu verschwinden, jedesmal eine Menge Energie, so daß die Batterien in Ihrem Transmitter schnell verbraucht sind. Ihre Energie reicht für fünfmaliges Beamen aus.

Ferner befinden sich in diesem Labyrinth Fallen für den Geist. Diese haben die gleiche Funktion wie Ihr Transmitter, allerdings ohne Energieverlust.

Versuchen Sie deshalb bei Ihrem Weg durch das Labyrinth den Geist in diese Fallen zu locken, um mit dem Aufsammeln der Gold-Dollar etwas Zeit zu gewinnen. Benutzen Sie nur in aussichtsloser Position Ihren Transmitter.

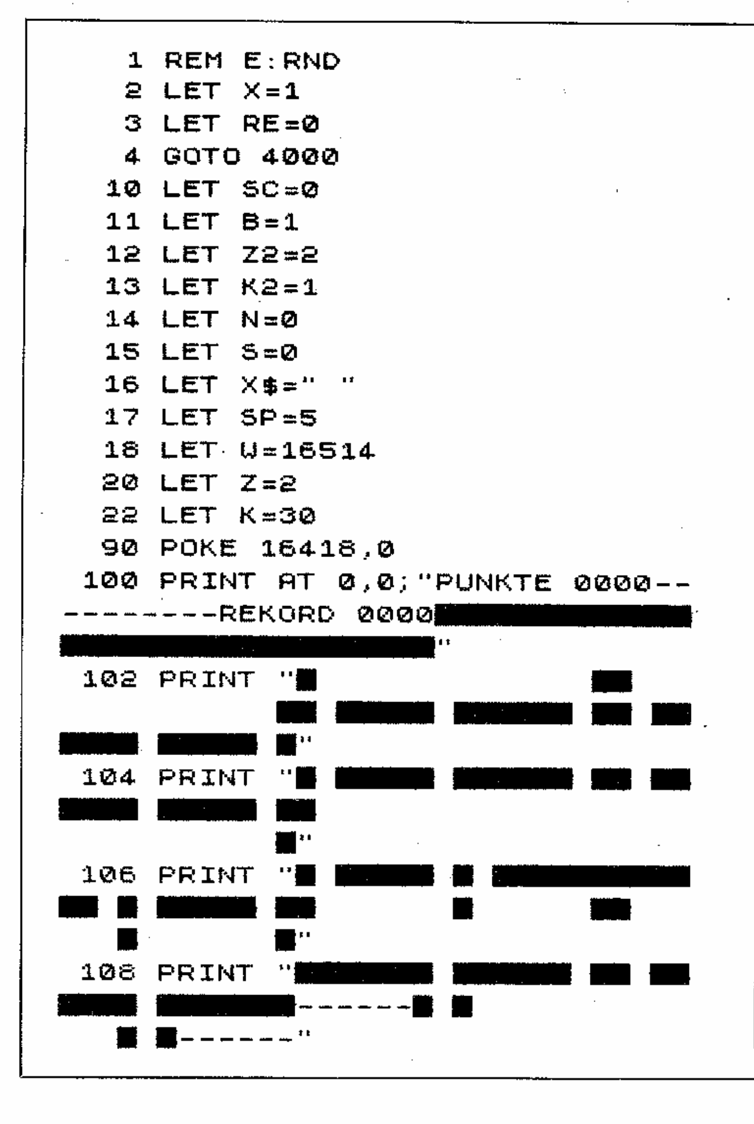
Jedesmal bei Spielbeginn werden die Gold-Dollars und Fallen für den Geist per Zufallsgenerator festgesetzt. Somit müssen Sie sich nach jeder Spielrunde eine neue Strategie zurechtlegen, um Punkte zu sammeln.

Ihre erreichte Punktzahl wird in HI-Score gespeichert. Nur wenn Sie die Übersicht behalten und nicht in Panik geraten, haben Sie eine Chance gegen den Geist.

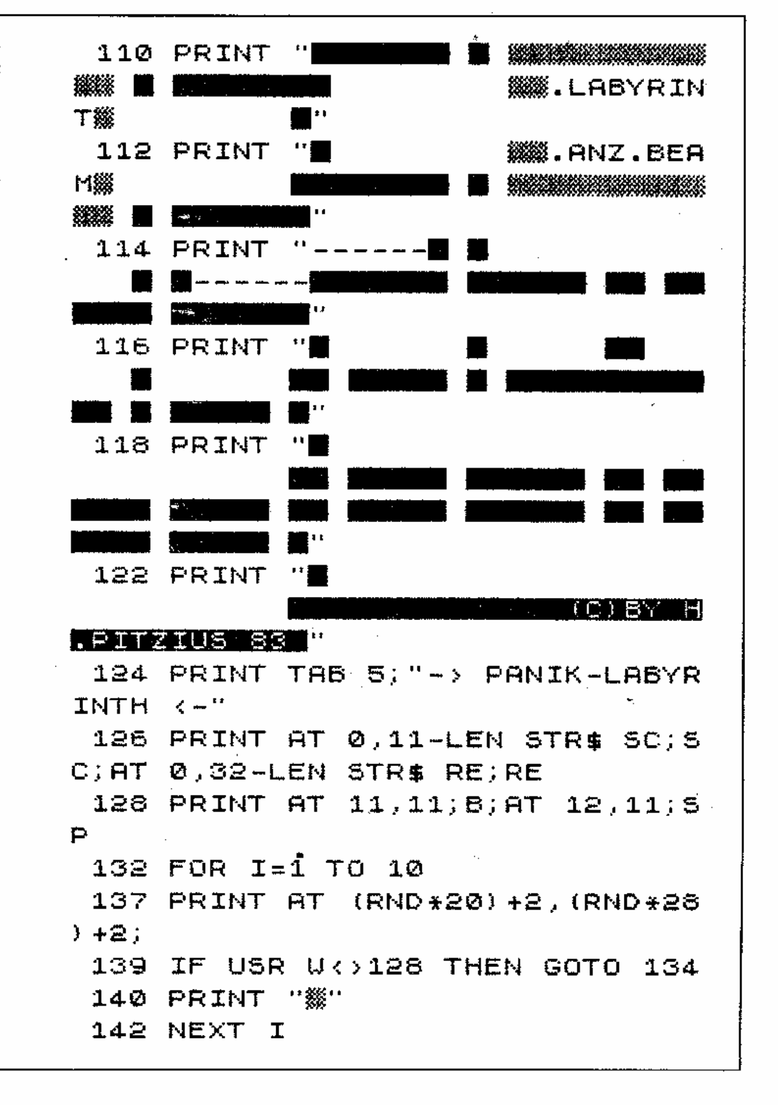
Üben Sie und lassen Sie Ihre Freunde und Bekannten versuchen, Ihren Rekord zu brechen.

Interessant für alle Programmierer dürfte die Maschinenprogrammroutine zur Bildschirmabfrage sein. Diese ersetzt die umständliche Abfrage der Basic-Routine mit Peek und erspart zudem noch eine Menge Speicherplatz. Außerdem ist sie natürlich wesentlich schneller als die Basic-Routine und kann sogar als AT- und TAB-Funktion nach vorheriger Position vor der Abfrage genutzt werden. Im Programmlisting finden Sie solch ein Beispiel in den Zeilen 415 bis 432.

Steuerung: Richtung über die Cursor-Tasten. Beamstrahl wird durch Taste ›9‹ aktiviert. Viel Spaß ...



```
   1 REM E:RND
   2 LET X=1
   3 LET RE=0
   4 GOTO 4000
  10 LET SC=0
  11 LET B=1
  12 LET Z2=2
  13 LET K2=1
  14 LET N=0
  15 LET S=0
  16 LET X$=" "
  17 LET SP=5
  18 LET W=16514
  20 LET Z=2
  22 LET K=30
  90 POKE 16418,0
 100 PRINT AT 0,0;"PUNKTE 0000----------REKORD 0000██████████████████████"
 102 PRINT "█         ██ ██ ████ ████ ██ ██ ██ ███ █"
 104 PRINT "█ ███ ████ ██ ██ ███ █"
 106 PRINT "█ ███ █ ██████ ██ █ ███ ██ █ █ █"
 108 PRINT "████ ████ ██ ██ ██ ████------█ █ █ █------"
```



```
 110 PRINT "█████ █ ▒▒▒▒▒▒▒▒▒ ▒▒ █ ██████ ▒▒.LABYRINT▒ █"
 112 PRINT "█ ▒▒.ANZ.BEAM▒ █████ █ ▒▒▒▒▒▒▒▒▒▒ ▒▒ █ █████"
 114 PRINT "------█ █ █ █------████ ████ ██ ██ ███ ████"
 116 PRINT "█ █ ██ █ ██ ████ █ ██████ ██ █ ████ █"
 118 PRINT "█ ██ ████ ████ ██ ██ ███ ████ ██ ████ ████ ██ ██ ███ ████ █"
 122 PRINT "█ ██████████(C)BY H.PITZIUS 83"
 124 PRINT TAB 5;"-> PANIK-LABYRINTH <-"
 126 PRINT AT 0,11-LEN STR$ SC;SC;AT 0,32-LEN STR$ RE;RE
 128 PRINT AT 11,11;B;AT 12,11;SP
 132 FOR I=1 TO 10
 137 PRINT AT (RND*20)+2,(RND*28)+2;
 139 IF USR W<>128 THEN GOTO 134
 140 PRINT "▒"
 142 NEXT I
```

```
 162 FOR I=1 TO 25
 166 PRINT AT (RND*21)+1,(RND*28
)+2;
 170 IF USR W<>0 THEN GOTO 163
 180 PRINT "*"
 190 NEXT I
 200 PRINT AT Z,K;"O"
 205 PRINT AT 0,13;"SPIEL."
 232 IF INKEY$="9" THEN GOSUB 33
0
 240 LET Z1=Z+(INKEY$="6")-(INKE
Y$="7")
 250 LET K1=K+(INKEY$="8")-(INKE
Y$="5")
 252 PRINT AT Z1,K1;
 260 IF USR W<>23 THEN GOTO 300
 270 LET N=N+1
 275 LET SC=SC+10
 290 GOTO VAL "310"
 300 IF USR W<>0 THEN GOTO 415
 305 LET SC=SC+1
 310 PRINT AT Z,K;" "
 312 PRINT AT Z1,K1;"O"
 320 IF N=25 THEN GOTO VAL "3000
"
 322 GOTO 400
 330 IF SP<1 THEN RETURN
 331 PRINT AT Z,K;" "
 332 LET Z=(RND*20)+1
 333 LET K=(RND*28)+1
 334 PRINT AT Z,K;
 335 IF USR W<>0 THEN GOTO 332
 337 LET SP=SP-1
 338 PRINT AT 12,11;SP
 339 RETURN
 400 LET Z=Z1
 410 LET K=K1
 415 PRINT AT Z2,K2;X$
 420 LET Z2=Z2+(Z>Z2)*1-(Z<Z2)*1
 430 LET K2=K2+(K>K2)*1-(K<K2)*1
 432 PRINT AT Z2,K2;
 450 IF USR W=52 THEN GOTO 2000
 456 LET X$=CHR$ USR W
 457 IF USR W=136 THEN GOSUB 560
 460 PRINT AT Z2,K2;"£"
 465 IF S=SC THEN GOTO VAL "210"
 466 LET S=SC
 470 LET S$=STR$ SC
 480 PRINT AT 0,11-LEN S$;S$
 540 GOTO VAL "210"
 550 GOTO VAL "600"
 560 LET Z2=INT (RND*18)+1
 562 LET K2=INT (RND*28)+1
 563 PRINT AT Z2,K2;
 565 IF USR W<>CODE " " THEN GOT
O VAL "560"
 566 PRINT AT Z2,K2;"£"
 567 LET X$=" "
 568 RETURN
 600 IF SC>RE THEN LET RE=SC
```



LASER 110 – 4 KByte RAM. Dieser „persönliche Computer" eignet sich als Helfer beim Studium, am Arbeitsplatz oder bei Analysen und Statistiken. Mit seinen Peripheriegeräten erfüllt er alle Ansprüche an einen Home-Computer.

```
680 PRINT AT 22,4;"      NOCHMAL
? (J/N)        "
690 IF INKEY$<>"" THEN GOTO VAL
"690"
700 IF INKEY$="" THEN GOTO VAL
"700"
710 IF INKEY$="N" THEN STOP
720 POKE 16418,2
730 GOTO VAL "10"
1000 LET Z2=2
1010 LET K2=2
1020 RETURN
2000 FOR I=0 TO 20
2002 PRINT AT Z2,K2;"█"
2004 PRINT AT Z2,K2;"£"
2006 NEXT I
2008 FOR I=0 TO 10
2009 PRINT AT 22,4;"SIE SIND ERW
ISCHT WORDEN"
2010 PRINT AT 22,4;"SIE SIND ERW
ISCHT WORDEN"
2020 NEXT I
2030 PRINT AT 0,11-LEN STR$ SC;S
C
2040 FOR I=1 TO 50
2050 NEXT I
2060 GOTO VAL "600"
3000 PRINT AT 22,4;"BRAVO ALLES
EINGESAMMELT"
3002 LET B=B+1
3010 PRINT AT 0,11-LEN STR$ SC;S
C
3020 FOR I=1 TO 50
3030 NEXT I
3040 GOTO VAL "12"
4000 LET C$="▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
4002 LET C$=C$+" [graphics characters]"
4004 LET C$=C$+"▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒"
4006 PRINT AT 8,0;
4008 FOR F=1 TO LEN C$
4010 IF INKEY$<>"" THEN GOTO 500
0
4012 PRINT C$(F);
4014 NEXT F
4015 PRINT AT 21,16;"(C)BY H.PIT
ZIUS"
4016 PRINT AT 8,0;
4018 FOR F=1 TO LEN C$
4020 PRINT " ";
4022 IF INKEY$<>"" THEN GOTO 500
0
4024 NEXT F
4026 GOTO 4000
5000 CLS
5002 PRINT " P A N I K - L A B Y
 R I N T H  ▀"▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀
▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀"
5004 PRINT ,,"IHRE AUFGABE IST E
S,IN EINEM LA-BYRINTH ALLE GOLDB
ARREN >*< ZU  SAMMELN."
5006 PRINT "DOCH ZU IHREM >O< UE
BEL,DAS LA- BYRINTH WIRD VON EIN
EM GEIST >£<BEWACHT,DER AUCH DUR
CH DIE MAUERGEHEN KANN."
5008 PRINT "DAMIT IHR UNTERNEHME
N NICHT AUS-SICHTSLOS IST,SIND F
ALLEN >▒<   FUER DEN GEIST AUFGE
BAUT."
5010 PRINT "GERAET DER GEIST >£<
 IN EINE     DER FALLEN >▒< WIRD
ER IM LABY- RINTH UMHERGESCHLEUD
ERT."
5012 PRINT AT 21,0;"DRUECKEN SIE
 EINE TASTE....."
5014 IF INKEY$="" THEN GOTO 5014
5016 CLS
5018 PRINT "IHRE EINZIGE WAFFE I
ST EIN      BEAMSTRAHL,DER SIE I
NNERHALB SE-KUNDEN AUS DER GEFAH
RENZONE BE- FOERDERT."
5020 PRINT "DER BEAMSTRAHL FUNKT
IONIERT JE- DOCH NUR,WENN SIE UE
BER GENUE-  GEND ENERGIE VERFUEG
EN."
5022 PRINT ,,"STEUERUNG:"
5024 PRINT ,,"RICHTUNG UEBER DIE
 KORSUR-TAS-  TEN."
5026 PRINT "BEAMSTRAHL WIRD DURC
H TASTE >9< AKTIVIERT."
5028 PRINT ,,"VIEL SPASS...."
5030 PRINT AT 21,0;"DRUECKEN SIE
 NUN EINE TASTE...."
5032 IF INKEY$="" THEN GOTO 5032
5034 CLS
5036 GOTO 5
9000 SAVE "PANIK"
9002 GOTO 1
9004 STOP
9010 REM BILDSCHIRMABFRAGE
9100 REM 042 014 126 079 006 000
         201
9110 REM BITTE IN ERSTE REM-
9115 REM ZEILE POKEN
```



Ein Maskengenerator für den TI 99/4 A in Extended BASIC geschrieben

DISPLAY AT (Zeile, Spalte), SIZE (Länge), BEEP: USING „##.##", ACCEPT AT (Zeile, Spalte), SIZE (Länge), VALIDATE (Typ), BEEP: Welcher TI 99/4 A-Besitzer kennt nicht diese mühselige Prozedur der Maskenprogrammierung?

Solange die Eingabefelder immer in der gleichen Spalte liegen, und die Variablen alle vom gleichen Typ sind, kann man sich die Arbeit mit einer Schleife noch erleichtern. Werden jedoch die Felder an verschiedenen Positionen gebraucht, und sind die Variablen auch noch von verschiedenen Typen, so geht die große Zählerei auf dem Bildschirm los. Welcher TI-Besitzer hat sich in solchen Fällen nicht schon einmal ein Programm gewünscht, welches ihm diese mühselige und stupide Arbeit abnimmt? Bitte schön: Hier ist Ihr gewünschtes Programm oder besser gesagt das Programmpaket: SCREEN DESIGNER.

Das Programm wurde auf dem TI 99/4 A in TI Extended BASIC geschrieben, und erfordert einen Kassettenrecorder, oder ein Diskettenlaufwerk, um die Maske als FILE zu speichern.

Das Programm besteht aus drei Teilen:

Teil 1: SCREEN DESIGNER

Dieses Programm ist das Kernstück des Paketes. Es beinhaltet den Maskengenerator, sowie den Editor zum ändern einer bereits erstellten Maske.

Teil 2: SCREEN READER

Hierbei handelt es sich um das Unterprogramm MASKE, welches die mittels SCREEN DESIGNER erstellte Maske später in Ihrem BASIC-Programm verwaltet.

Teil 3: SCREEN PRINTER

Für dieses Programm brauchen Sie außer der oben schon genannten Peripherie zusätzlich noch einen Drucker, denn dieses Programm erstellt Ihnen eine Aufschlüsselung einer mit SCREEN DESIGNER erstellten Maske, und gibt diese auf dem Drucker aus. So haben Sie die entsprechende Maske immer griffbereit bei Ihren Programmunterlagen, und können jederzeit den Aufbau und die Gliederung der Maske vergleichen.

```
100 !@P-
110 REM *******************
120 REM * SCREEN  DESIGNER *
130 REM *-----------------*
140 REM *   TI 99 / 4 A   *
150 REM * + Extended BASIC *
160 REM * + Ext. Speicher  *
170 REM *-----------------*
180 REM *(c)1984 Tronicsoft*
190 REM *        V.Becker  *
200 REM *******************
210 REM
220 REM
230 !@P+
```

Um die Verwendung des Programmes SCREEN DESIGNER möglichst vielen TI-Benutzern zugänglich zu machen, wurde hier die Form eines FILES zur Speicherung der MASKE gewählt. Es wäre ohne weiteres möglich gewesen, die Maske auch als Programm erstellen zu lassen, dies setzt jedoch ein Diskettenlaufwerk voraus, um das Programm im MERGE-Modus erstellen zu lassen. Da die meisten TI-User jedoch kein Laufwerk zur Verfügung haben, ist der hier beschriebene Lösungsweg sicherlich am besten.

Doch nun zum eigentlichen Programm:

Wir beginnen eine Maske zu erstellen, indem wir das Programm SCREEN DESIGNER einladen (z. B. RUN „CS" oder RUN „DSK1.S-DESIGNER"). Das Programm startet nun von selbst, und das Titelbild erscheint. Sie haben jetzt zwischen zwei Möglichkeiten die Auswahl: 1. Sie wollen eine neue Maske erstellen oder 2. Sie wollen eine bereits erstellte Maske, die bereits auf Cassette oder Diskette gespeichert ist, ändern.

Bei Auswahl 2 werden Sie nun nach dem Namen der zu ändernden Maske gefragt. Geben Sie den Namen ein (z. B. CS1 oder DSK1.MASKE) und drücken Sie <ENTER>. Der Rechner lädt nun die gewünschte Maske ein, und gibt diese auf dem Leerbildschirm aus. Wenn Sie die Auswahl 1 getroffen haben, so geht der Rechner sofort in den Leerbildschirm, ohne etwas auszugeben.

Der Leerbildschirm hat eine gelbe Hintergrundfarbe und einen blauen Rand. In der Ecke ist ein L sichtbar. Bewegen Sie den in der oberen linken Ecke sichtbaren Cursor mittels den Cursor-Steuertasten über den Bildschirm. Schreiben Sie mittels der Tastatur Ihren Text an die gewünschten Stellen auf dem Bildschirm, oder überschreiben Sie den entsprechenden Text. Sie können an den Stellen, wo später die Eingabefelder liegen sollen, beliebige Zahlen oder Buchstaben eingeben. Vergessen Sie jedoch nicht die Dezimalstelle mit einem Punkt zu versehen, da der Rechner sonst bei der Generierung des USING-Formates keine Dezimalstellen generiert. Eine Betätigung der <ENTER>-Taste setzt den Cursor an die linke obere Bildschirmecke zurück. Nachdem Sie sich so Ihre Maske definiert haben, drücken Sie die Funktionstaste AID (FCTN 7), um den Leerbildschirm zu verlassen. Der Rechner speichert jetzt die Maske ab. Dieser Vorgang dauert ca. 50 sec. Zum Zeichen, daß das System arbeitet, wird der Rand des Bildschirmes rot.

Sobald die Maske abgespeichert ist, gelangen Sie in den Attributbildschirm. Der Rand hat jetzt die Farbe Weiß, und in der linken oberen Ecke ist ein A zu sehen.

Im Attributbildschirm wird die Lage der Ein- bzw. Ausgabefelder festgelegt. Wenn Sie eine bereits gespeicherte Maske editieren, so werden die Attribute der ursprünglichen Maske automatisch gesetzt. Bei einer Neudefinition erscheinen keine Attribute auf dem Bildschirm.

Die Attribute sehen wie folgt aus:

< Beginn eines Feldes

> Ende eines Feldes

Bewegen Sie den Cursor mit den Steuertasten an die gewünschte Stelle, und geben Sie dann das Attribut ein. Es können mit den Attributen auch Textstellen überschrieben werden. Dies macht der Maske nichts aus, da der Text bereits gespeichert ist, und durch ein Überschreiben nicht beeinträchtigt wird. Das entsprechende Feld wird immer zwischen Feldanfang- und Feldende-Attribut generiert. Die maximale Feldlänge darf jedoch 14 Zeichen nicht überschreiten. Ein Attribut kann durch die Funktionstaste ERASE (FCTN 3) gelöscht werden.

Es können maximal 25 Felder in einer Maske generiert werden.

Nachdem alle Attribute gesetzt wurden, verlassen Sie den Attributbildschirm wieder mit der Funktionstaste AID (FCTN 7). Jetzt führt der Rechner eine Fehlerkontrolle der Attribute durch.

Sollte ein Fehler aufgetreten sein, so meldet sich der Rechner mit einem akustischen Signal, und in der oberen linken Ecke erscheint ein F. Der Cursor hat jetzt die Farbe Weiß und steht an der fehlerhaften Bildschirmstelle.




```
240 DIM MASKE$(24),VALD$(25),FELD(25),FA(25),FE(25),USER$(25),TON(25)
250 GOTO 280 :: B$ :: DATEI$ :: VA$ :: X$ :: DEL :: I :: K :: MASKE :: S :: SA :
: SIGN :: SIGNALT :: SKIP :: SUCH :: T :: W1 :: W2 :: WA :: WE :: WERT :: X :: X
1 :: XS :: Y :: Y1 :: YS :: ZA
260 CALL CHAR :: CALL CLEAR :: CALL COLOR :: CALL DELSPRITE :: CALL GCHAR :: CAL
L HCHAR :: CALL KEY :: CALL MAGNIFY :: CALL SCREEN :: CALL SOUND :: CALL SPRITE
270 CALL VCHAR :: !@P-
280 FA(0),FE(0)=1
290 X=3 :: Y=1
300 GOTO 2070
310 CALL CLEAR :: CALL SCREEN(5):: CALL COLOR(1,2,5)
320 CALL CHAR(128,RPT$("0",16))
330 CALL CHAR(96,"FF818181818181FF")
340 FOR I=1 TO 2 :: CALL VCHAR(1,I,128,24):: CALL VCHAR(1,I+30,128,24):: NEXT I
350 CALL COLOR(1,2,11,2,2,11,3,2,11,4,2,11,5,2,11,6,2,11,7,2,11,8,2,11,9,2,11)
360 CALL COLOR(10,2,11,11,2,11,12,2,11)
370 ON SKIP GOTO 400,380
380 GOSUB 1430
390 FOR I=0 TO 24 :: MASKE$(I)="" :: NEXT I
400 CALL SPRITE(£2,76,16,8,8)
410 T=13 :: GOSUB 500
420 CALL KEY(0,T,S):: IF S=0 THEN 420
430 IF T=13 THEN GOSUB 500 ELSE 450
440 GOTO 420
450 IF T>7 AND T<12 THEN GOSUB 500 ELSE 470
460 GOTO 420
470 IF T=1 THEN 650
480 XS=X :: YS=Y :: SIGN=T :: T=9 :: GOSUB 500
490 CALL HCHAR(YS,XS,SIGN):: GOTO 420
500 X1=X :: Y1=Y
510 IF T=13 THEN X=3 :: Y=1 :: GOTO 530
520 ON T-7 GOSUB 580,550,610,630
530 CALL SPRITE(£1,96,2,Y*8-7,X*8-7)
540 RETURN
550 X=X+1 :: IF X>30 THEN X=3 :: Y=Y+1
560 IF Y>24 THEN Y=1
570 RETURN
580 X=X-1 :: IF X<3 THEN X=30 :: Y=Y-1
590 IF Y<1 THEN Y=24
600 RETURN
610 Y=Y+1 :: IF Y>24 THEN Y=1
620 RETURN
630 Y=Y-1 :: IF Y<1 THEN Y=24
640 RETURN
650 CALL DELSPRITE(ALL):: CALL SCREEN(9):: CALL COLOR(13,9,1,14,9,1)
660 FOR I=1 TO 24
670 FOR K=3 TO 30
680 CALL GCHAR(I,K,MASKE):: MASKE$(I)=MASKE$(I)&CHR$(MASKE)
690 NEXT K :: NEXT I
700 FOR I=1 TO 24 :: DISPLAY AT(I,1):" " :: NEXT I
710 CALL SCREEN(16):: CALL COLOR(13,16,1,14,16,1)
720 FOR I=1 TO 24 :: DISPLAY AT(I,1):MASKE$(I):: NEXT I
730 ON SKIP GOTO 740,1670
740 ! Attributbildschirm
750 X=3 :: Y=1
760 CALL SPRITE(£2,65,5,8,8)
770 T=13 :: GOSUB 500
780 CALL KEY(5,T,S):: IF T>7 AND T<12 THEN GOSUB 500
790 IF T=60 XOR T=62 THEN 830
800 IF T=13 THEN CALL SOUND(-100,262,5):: GOSUB 500
810 IF T=7 THEN 860
820 IF T=1 THEN 870 ELSE 780
```

```
830 CALL HCHAR(Y,X,T)
840 T=9 :: CALL SOUND(-100,523,5):: GOSUB 500
850 GOTO 780
860 CALL HCHAR(Y,X,32):: SIGNALT=32 :: T=9 :: CALL SOUND(-100,131,5):: GOSUB 500
 :: GOTO 780
870 CALL DELSPRITE(ALL):: CALL COLOR(13,9,1,14,9,1):: CALL SCREEN(9)
880 SUCH=1
890 SKIP=0
900 FOR Y=1 TO 24
910 FOR X=3 TO 30
920 CALL GCHAR(Y,X,WERT)
930 IF WERT=60 OR WERT=62 THEN GOSUB 1810
940 IF SKIP=1 THEN GOTO 1770
950 NEXT X
960 NEXT Y
970 IF FA(SUCH)<>0 THEN 1770
980 CALL CLEAR :: CALL VCHAR(1,1,128,48):: CALL VCHAR(1,31,128,48):: CALL COLOR(
13,2,9,14,2,9):: PRINT "<S-D> generiert die Felder": :
990 CALL SOUND(-100,523,5)
1000 FOR SUCH=1 TO 25
1010 IF FA(SUCH)=0 THEN 1090
1020 WA=FA(SUCH)-INT(FA(SUCH)):: WA=WA*100 :: WE=FE(SUCH)-INT(FE(SUCH)):: WE=WE*
100
1030 USER$(SUCH)=SEG$(MASKE$(INT(FA(SUCH))),WA,FELD(SUCH))
1040 B$=USER$(SUCH)
1050 FOR K=1 TO LEN(USER$(SUCH))
1060 IF ASC(B$)=35 XOR ASC(B$)=46 THEN 1070 ELSE GOSUB 1760
1070 B$=SEG$(B$,2,LEN(B$))
1080 NEXT K :: NEXT SUCH
1090 PRINT "<S-D> loescht die Felder in       der Bildmaske": : :: CALL SOUND(-1
00,523,5)
1100 FOR SUCH=1 TO 25
1110 IF FA(SUCH)=0 THEN 1160
1120 W1=INT(FA(SUCH)):: W2=FA(SUCH)-INT(FA(SUCH)):: W2=W2*100
1130 FOR DEL=1 TO FELD(SUCH)
1140 MASKE$(W1)=SEG$(MASKE$(W1),1,W2-2+DEL)&" "&SEG$(MASKE$(W1),W2+DEL,28)
1150 NEXT DEL :: NEXT SUCH
1160 CALL CLEAR :: CALL COLOR(13,16,2,14,16,2):: CALL SCREEN(2):: CALL VCHAR(1,1
,128,48):: CALL VCHAR(1,31,128,48)
1170 CALL HCHAR(4,3,128,28)
1180 CALL HCHAR(21,3,128,28)
1190 FOR SUCH=1 TO 25
1200 IF FA(SUCH)=0 THEN 1410
1210 DISPLAY AT(2,1):MASKE$(INT(FA(SUCH))):: DISPLAY AT(2,(FA(SUCH)-INT(FA(SUCH)
))*100)SIZE(FELD(SUCH)):USER$(SUCH)
1220 DISPLAY AT(6,1):"Felder vor Ausgabe loeschen (J/N)   --->   <---"
1230 DISPLAY AT(9,1):"Akkustisches Signal bei der Ausgabe des Feldes          (J
/N)   --->   <---"
1240 DISPLAY AT(23,7):USING "Feld Nr.: ££":SUCH
1250 IF SGN(FELD(SUCH))=-1 THEN DISPLAY AT(7,14)SIZE(1):"N" ELSE DISPLAY AT(7,14
)SIZE(1):"J"
1260 ACCEPT AT(7,14)VALIDATE("JN")SIZE(-1)BEEP:X$ :: IF X$="N" THEN FELD(SUCH)=-
FELD(SUCH)
1270 IF TON(SUCH)=1 THEN DISPLAY AT(11,14)SIZE(1):"J" ELSE DISPLAY AT(11,14)SIZE
(1):"N"
1280 ACCEPT AT(11,14)VALIDATE("JN")SIZE(-1)BEEP:X$ :: IF X$="J" THEN TON(SUCH)=1
 ELSE TON(SUCH)=0
1290 DISPLAY AT(13,1):"Typ der Variablen          0 - keine Vereinbarung"
1300 DISPLAY AT(15,1):"1 - UALPHA (Grossbuchstaben)2 - DIGIT  (Ziffern 1 bis 9)3
 - NUMERIC(Ziffen,' .+-E ')4 - eigener Text"
1310 DISPLAY AT(19,9):"--->   <---"
1320 IF LEN(VALD$(SUCH))>1 THEN DISPLAY AT(19,14)SIZE(1):"4" :: VA$=VALD$(SUCH)E
```



```
LSE DISPLAY AT(19,14)SIZE(1):VALD$(SUCH)
1330 ACCEPT AT(19,14)SIZE(-1)VALIDATE("01234")BEEP:VALD$(SUCH)
1340 IF VALD$(SUCH)="" OR VALD$(SUCH)=" " THEN 1330
1350 IF VALD$(SUCH)="4" THEN 1360 ELSE 1390
1360 DISPLAY AT(19,1):"Text : "
1370 DISPLAY AT(19,9):VA$
1380 ACCEPT AT(19,9)SIZE(-20)BEEP:VALD$(SUCH)
1390 FOR I=13 TO 19 :: DISPLAY AT(I,1):"" :: NEXT I
1400 NEXT SUCH
1410 GOSUB 1430
1420 GOTO 1540
1430 CALL CLEAR
1440 CALL VCHAR(1,31,128,96)
1450 FOR I=1 TO 24
1460 DISPLAY AT(I,1):MASKE$(I)
1470 NEXT I
1480 FOR SUCH=1 TO 25
1490 IF FA(SUCH)=0 THEN 1530
1500 W1=FA(SUCH)-INT(FA(SUCH)):: W1=W1*100 :: W2=INT(FA(SUCH))
1510 DISPLAY AT(W2,W1)SIZE(FELD(SUCH)):USER$(SUCH)
1520 NEXT SUCH
1530 RETURN
1540 CALL KEY(5,T,S):: IF T<>15 THEN IF T<>12 THEN 1540
1550 IF T=15 THEN SKIP=2 :: GOTO 310
1560 CALL CLEAR :: DISPLAY AT(12,1):"Name der Datei, worauf die  Maske gespeiche
rt werden     soll ?  (Geraet.Name)"
1570 ACCEPT AT(15,1)BEEP:DATEI$
1580 OPEN £2:DATEI$,OUTPUT,SEQUENTIAL,INTERNAL,FIXED 80
1590 FOR I=1 TO 24 :: PRINT £2:MASKE$(I):: NEXT I
1600 FOR I=1 TO 25 :: PRINT £2:FELD(I),TON(I),FA(I),VALD$(I),USER$(I)
1610 IF FA(I)=0 THEN 1630
1620 NEXT I
1630 CALL CLEAR
1640 ON ERROR STOP
1650 CLOSE £2
1660 STOP
1670 FOR I=1 TO 25
1680 IF FA(I)=0 THEN 1730
1690 ZA=INT(FA(I)):: SA=(FA(I)-INT(FA(I)))*100
1700 DISPLAY AT(ZA,SA-1)SIZE(1):"<"
1710 DISPLAY AT(ZA,SA+ABS(FELD(I)))SIZE(1):">"
1720 NEXT I
1730 FOR I=1 TO 25 :: FA(I),FE(I)=0 :: USER$(I)="" :: NEXT I
1740 FA(0),FE(0)=1
1750 GOTO 740
1760 USER$(SUCH)=SEG$(USER$(SUCH),1,K-1)&"£"&SEG$(USER$(SUCH),K+1,LEN(USER$(SUCH
))-K):: RETURN
1770 CALL SPRITE(£2,70,16,8,8)
1780 CALL SPRITE(£1,96,16,Y*8-7,X*8-7):: CALL SOUND(1000,262,0,330,0,196,0)
1790 FOR I=1 TO 25 :: FA(I),FE(I),FELD(I)=0 :: NEXT I
1800 GOTO 780
1810 !
1820 IF WERT=62 THEN 1860
1830 IF FA(SUCH)<>0 THEN SKIP=1 :: RETURN
1840 FA(SUCH)=Y+(X-1)/100
1850 RETURN
1860 IF FA(SUCH)=0 THEN SKIP=1 :: RETURN
1870 IF INT(FA(SUCH))<>Y THEN SKIP=1 :: RETURN
1880 FE(SUCH)=Y+(X-2)/100
1890 W1=FE(SUCH)-INT(FE(SUCH)):: W1=W1*100 :: W2=FA(SUCH)-INT(FA(SUCH)):: W2=W2*
100
1900 FELD(SUCH)=W1-W2
```

```
1910 IF FELD(SUCH)<1 OR FELD(SUCH)>14 THEN SKIP=1 :: RETURN
1920 SUCH=SUCH+1
1930 IF SUCH>25 THEN SKIP=1 :: RETURN
1940 !
1950 RETURN
1960 DISPLAY AT(15,1):"Welche Maske soll veraendertwerden ?   (Geraet.name)"
1970 ACCEPT AT(17,1)BEEP:DATEI$
1980 ON ERROR 1970
1990 OPEN £1:DATEI$,INPUT ,SEQUENTIAL,INTERNAL,FIXED 80
2000 FOR I=1 TO 24 :: INPUT £1:MASKE$(I):: NEXT I
2010 FOR I=1 TO 25 :: INPUT £1:FELD(I),TON(I),FA(I),VALD$(I),USER$(I)
2020 IF FA(I)=0 THEN 2040
2030 NEXT I
2040 CLOSE £1
2050 ON ERROR STOP
2060 GOTO 2170
2070 FOR I=2 TO 12 :: CALL COLOR(I,2,11):: NEXT I
2080 CALL CLEAR :: CALL VCHAR(1,31,128,96):: CALL COLOR(1,2,11,13,11,5):: CALL S
CREEN(5)
2090 CALL MAGNIFY(2):: CALL SPRITE(£1,83,7,17,60,£2,68,7,17,132)
2100 DISPLAY AT(4,8):"creen    esigner"
2110 DISPLAY AT(23,4):"(c) 1984 by Tronicsoft"
2120 DISPLAY AT(9,1):"Bitte waehlen Sie aus --->"
2130 DISPLAY AT(15,1):"1 - Neue Maske erstellen"
2140 DISPLAY AT(17,1):"2 - Vorhandene Maske aendern"
2150 ACCEPT AT(9,28)VALIDATE("12")BEEP:SKIP
2160 ON SKIP GOTO 2170,1960
2170 CALL DELSPRITE(ALL):: CALL CLEAR :: CALL MAGNIFY(1):: GOTO 310
```

Unterprogramm SCREEN READER

Um eine mit SCREEN DESIGNER erstellte Maske im eigenen Extended-BASIC-Programm zu verwenden, wird das Unterprogramm MASKE, der sogenannte SCREEN READER benutzt.

Benutzer eines Kassettenrecorders schreiben das Programm separat auf Kassette weg, und laden dieses, bevor sie ihr eigenes Programm schreiben, ein. Benutzer eines Diskettenlaufwerkes speichern das Programm im MERGE Modus ab (z. B. SABE DKS1.MASKE,MERGE) und können dieses dann mit MERGE auf jedes beliebige Programm aufladen. Dabei ist lediglich zu beachten, daß die Programmzeilen ab 32000 nicht vom eigenen Programm belegt sind.

Doch nun zum Programm:

Zeile 32060 hat die allgemeine Form:
DIM MASKE$(n,24), VALD$(n,25), SZE(n,25), DAT(n,25), USER$(n,25), Ton(n,25)
wobei n die Anzahl der gleichzeitig verwaltbaren Masken angibt. Die Anzahl ist lediglich durch den zur Verfügung stehenden Speicherplatz begrenzt.

Im Programm ist n mit 1 angegeben. Das bedeutet, daß maximal 2 Masken gleichzeitig verwaltet werden können (Maske Nr. 0 und Maske Nr. 1). Die Anzahl kann jedoch vom Benutzer beliebig erhöht werden.

Der Aufruf des Unterprogramms hat folgende Form:
CALL MASKE (ACT,MNR,FNR, numerische Variable oder Zahl, Stringvariable oder Text).

Hierin bedeuten:

ACT – Eine Anzahl zwischen 1 und 5, welche angibt, was mit, bzw. in der Maske geschehen soll:

1 – Die Maske wird auf dem Bildschirm ausgegeben.

2 – Eine Variable wird in der Maske eingelesen.

3 – Eine Variable, bzw. Zahl oder Text wird in einem Feld der Maske ausgegeben.

4 – Eine Maske wird von Kassette oder Diskette geladen.

5 – Das Unterprogramm wird für eine oder mehrere Masken initialisiert, d. h., es wird der Speicherplatz für die Masken zur Verfügung gestellt.

MNR – Gibt die Nummer der Maske an, mit welcher die entsprechende Aktion ausgeführt werden soll.

FNR – Gibt die Feldnummer in der entspr. Maske an, in welcher die Aktion durchgeführt werden soll.

Numerische Variable – ist eine Variable nach Ihrer Wahl.

Stringvariable – ist eine Textvariable nach Ihrer Wahl.

Die ganze Maskenverwaltung ist relativ einfach mit einem einzigen Befehl zu bearbeiten.

Wir wollen dies an einem Beispiel verdeutlichen:

Nehmen wir an, Sie haben 2 Masken, welche Sie verwalten wollen, und nehmen wir an, diese Masken sind auf Kassette gespeichert.

Zuerst müssen wir die Maskenverwaltung initialisieren und geben den Befehl: CALL MASKE(5,0,0,V,V$) ein. Die DIM-Anweisung in Zeile 32060 wurde nicht verändert, und der Platz für 2 Masken ist jetzt reserviert.

Dieser Befehl muß als erster Maskenbefehl stehen, und darf im weiteren Programmverlauf nicht noch einmal vorkommen.

Jetzt laden wir die erste Maske ein. Dazu geben wir den Befehl: CALL MASKE(4,0,0,V,V$) und der Rechner fragt nach dem Eingabegerät. Tippen Sie CSi ein, und drücken Sie danach ⟨ENTER⟩. Der Rechner lädt die Maske unter der Nummer 0 ein.

Zum Einlesen der 2. Maske geben wir den Befehl CALL MASKE(4,1,0,V,V$) ein, und wieder werden wir nach dem Eingabegerät gefragt. Geben Sie wieder CSi ein und drücken Sie ⟨ENTER⟩, um die 2. Maske zu laden. Die beiden Masken sind nun im Rechner gespeichert.



Wir rufen jetzt Maske 2 auf: CALL MASKE(1,1,0,V,V$), und die Maske wird auf dem Bildschirm ausgegeben. Nehmen wir an, die Maske hat 15 Felder, und aus allen Feldern soll eine Variable eingelesen werden. Dazu haben wir zuvor im Programm ein Feld mit dem Namen EINGABE dimensioniert.
Die Eingabe der Felder kann nun wie folgt erfolgen:
CALL MASKE(2,1,I,EINGABE(I),V$),
NEXT I.
Sie haben somit alle 15 Felder der Maske eingegeben, wobei es völlig egal ist, wo die Felder auf dem Bildschirm liegen.
Der SCREEN READER vereinbart die Felder wie folgt:
Felder, die den Variablentyp DIGIT oder NUMERIC haben, werden in die numerische Variable eingelesen.
Felder mit den Variablentypen UALPHA, eigener Text oder wenn keine Vereinbarung getroffen wurde, werden in die Stringvariable des Parameters eingelesen.

Doch nun weiter im Beispiel:

Wir geben jetzt unsere Maske 1 auf dem Bildschirm mit dem Befehl CALL MASKE(1,0,0,V,V$) aus.
Das Feld 3 haben wir im SCREEN DESIGNER als Stringfeld ausgewiesen, und wollen hier den Kommentar Computronic ausgeben. Dazu schreiben wir den Befehl CALL MASKE(3,0,3,V,„Computronic") und der Text erscheint im Feld 3.
Die Maskenverwaltung geht mit diesem Unterprogramm wesentlich komfortabler und einfacher als mit dem ewigen DISPLAY AT und ACCEPT AT. Die Verarbeitung ist weiterhin auch platzsparender, da man im Verlauf eines Programmes jederzeit eine nicht mehr benötigte Maske durch eine andere Maske überschreiten kann.
Nehmen wir an, wir brauchen unsere Maske 2 nicht mehr, so können wir mit CALL MASKE(4,1,0,V,V$) eine neue Maske in den Speicherbereich der Maske 2 laden. Wird dann die Maske 2 angesprochen, so erhalten wir die neu eingeladene Maske auf den Bildschirm.
Es gibt viele Möglichkeiten der Maskenverwaltung, die in dieser kurzen Beschreibung nicht alle abgehandelt werden können. Am besten Sie probieren die Sache einfach selbst aus.
Es ist jedoch darauf zu achten, daß sie keine Maskennummer ansprechen, die außerhalb der durch DIM in Zeile 32060 liegenden Dimension liegt. Ferner dürfen Sie keine Feldnummer ansprechen, die nicht im SCREEN DESIGNER definiert wurde.
Ein Versuch dieses zu tun, wird mit einer Fehlermeldung und mit Programmbruch quittiert.
Es gehört sicher etwas Übung zu dieser neuartigen Programmtechnik, jedoch werden Sie dieses ganz schnell in den Griff bekommen, und dann den Maskengenerator nicht mehr missen wollen.
Mit dem Programm SCREEN PRINTER können Sie sich eine Aufschlüsselung einer durch SCREEN DESIGNER erstellten Maske auf dem Drukker ausdrucken lassen. Sie haben somit mit einem Blick die Kontrolle, wie die Maske aufgebaut ist, und wo die einzelnen Felder liegen. Ferner bekommen Sie sämtliche Optionen der einzelnen Felder ausgedruckt.
Als Anlage ist ein Probeausdruck einer Maske beigefügt.
Die im Feld „Länge" ausgebenen Werte sind wie folgt zu lesen:
Postitive Länge – Das Feld wird vor Eingabe gelöscht.
Negative Länge – Das Feld wird nicht gelöscht.
Laden Sie das Programm mit RUN „CS1" oder mit RUN „DSK1.S-PRINTER" ein. Es startet sich jetzt von selbst.
Geben Sie den Namen der aufzuschlüsselnden Maske ein (z. B. CS1 oder DSK1.MASKE).
Sie werden jetzt nach dem Druckertyp gefragt. Hierbei antworten Sie dem Rechner mit der entsprechenden Schnittstellenoption, an welche Sie Ihren Drucker angeschlossen haben (z. B. RS232 für die serielle Schnittstelle, oder PIO für die parallele Schnittstelle).
Danach geben Sie dem Rechner noch das aktuelle Datum ein, und der Ausdruck beginnt.

ANMERKUNG:

Das Programm ist auf einen SEIKOSHA-Drucker der GP-Serie abgestimmt. Sollten Sie einen anderen Drucker besitzen, so müssen Sie möglicherweise die Programmzeile 620 Print #1:CHR$(15) abändern, denn dieses Steuerzeichen schaltet den Drucker von Breitschrift auf Normalschrift zurück.
So muß die Zeile z. B. bei Verwendung eines STAR Druckers PRINT #1: CHR$(20) heißen.

```
32000 SUB MASKE(ACT,MNR,FNR,VAR,VAR$)
32010 REM UnterProgramm SCREEN READER
32020 REM Maskenverwaltungsprogramm zu SCREEN DESIGNER
32030 REM (c) 1984 by Tronicsoft
32040 STV=0
32050 ON ACT GOTO 32370,32090,32110,32270,32060
32060 DIM MASKE$(1,24),VALD$(1,25),SZE(1,25),DAT(1,25),USER$(1,25),TON(1,25)
32070 ON WARNING NEXT
32080 SUBEXIT
32090 STV=1
32100 IF TON(MNR,FNR)=1 THEN CALL SOUND(-50,698,0)
32110 IF LEN(VALD$(MNR,FNR))>1 THEN 32210
32120 IF VAL(VALD$(MNR,FNR))=0 THEN 32210
32130 ON VAL(VALD$(MNR,FNR))GOTO 32210,32140,32140
32140 DISPLAY AT(INT(DAT(MNR,FNR)),(DAT(MNR,FNR)-INT(DAT(MNR,FNR)))*100)SIZE(SZE
(MNR,FNR)):USING USER$(MNR,FNR):VAR
32150 IF STV=0 THEN SUBEXIT
32160 ON VAL(VALD$(MNR,FNR))GOTO 32390,32170,32190
32170 ACCEPT AT(INT(DAT(MNR,FNR)),(DAT(MNR,FNR)-INT(DAT(MNR,FNR)))*100)SIZE(SZE(
MNR,FNR))VALIDATE(DIGIT):VAR
32180 SUBEXIT
```

```
32190 ACCEPT AT(INT(DAT(MNR,FNR)),(DAT(MNR,FNR)-INT(DAT(MNR,FNR)))*100)SIZE(SZE(
MNR,FNR))VALIDATE(NUMERIC):VAR
32200 SUBEXIT
32210 DISPLAY AT(INT(DAT(MNR,FNR)),(DAT(MNR,FNR)-INT(DAT(MNR,FNR)))*100)SIZE(SZE
(MNR,FNR)):USING USER$(MNR,FNR):VAR$
32220 IF STV=0 THEN SUBEXIT
32230 IF VALD$(MNR,FNR)="1" THEN ACCEPT AT(INT(DAT(MNR,FNR)),(DAT(MNR,FNR)-INT(D
AT(MNR,FNR)))*100)SIZE(SZE(MNR,FNR))VALIDATE(UALPHA):VAR$
32240 IF VALD$(MNR,FNR)="0" THEN ACCEPT AT(INT(DAT(MNR,FNR)),(DAT(MNR,FNR)-INT(D
AT(MNR,FNR)))*100)SIZE(SZE(MNR,FNR)):VAR$
32250 IF LEN(VALD$(MNR,FNR))>1 THEN ACCEPT AT(INT(DAT(MNR,FNR)),(DAT(MNR,FNR)-IN
T(DAT(MNR,FNR)))*100)SIZE(SZE(MNR,FNR))VALIDATE(VALD$(MNR,FNR)):VAR$
32260 SUBEXIT
32270 CALL CLEAR :: DISPLAY AT(1,1)BEEP:"Name der zu ladenden Maske :(Geraet.Dat
ei)" :: ACCEPT AT(3,1):DATEI$
32280 ON ERROR 32270
32290 OPEN £99:DATEI$,INPUT ,SEQUENTIAL,INTERNAL,FIXED 80
32300 FOR I=1 TO 24 :: INPUT £99:MASKE$(MNR,I):: NEXT I
32310 FOR I=1 TO 25 :: INPUT £99:SZE(MNR,I),TON(MNR,I),DAT(MNR,I),VALD$(MNR,I),U
SER$(MNR,I)
32320 IF DAT(MNR,I)=0 THEN 32340
32330 NEXT I
32340 CLOSE £99 :: CALL CLEAR
32350 ON ERROR STOP
32360 SUBEXIT
32370 CALL CLEAR
32380 FOR I=1 TO 24 :: DISPLAY AT(I,1):MASKE$(MNR,I):: NEXT I
32390 SUBEND
```

Ein Fehler kann auftreten, wenn:

1. Ein Feldende Attribut gesetzt wurde, ohne daß ein Feldanfang Attribut gesetzt wurde.
2. Ein zweiter Feldanfang definiert wurde, ohne daß das vorherige Feld geschlossen wurde.
3. Ein Feld länger als 14 Zeichen ist.
4. Ein Feld in einer Zeile beginnt und erst in der nächsten Zeile endet.
5. Das letzte Feldende Attribut vergessen wurde. In diesem Fall erscheint der Cursor nicht, wird jedoch nach der Betätigung einer Cursorsteuertaste in der unteren rechten Ecke sichtbar.

Steuern Sie im Falle eines Fehlers den Cursor an die entsprechende Stelle, und ändern Sie diese, wie im Attributbildschirm beschrieben, ab. Danach drücken Sie wieder „AID", um einen neuen Fehlertestlauf zu starten.

Wenn keine Fehler im Attributbildschirm aufgetreten sind, so beginnt der Rechner mit der eigentlichen Maskengenerierung. Die Aktionen werden Ihnen auf dem Bildschirm mitgeteilt.

Der Rechner generiert in dieser Zeit das DISPLAY AT Format, die Feldlänge SIZE(), sowie das USING Format.

Nachdem die Generierung abgeschlossen ist, kommen Sie zur Variablenvereinbarung:

Im ersten Feld wird gefragt, ob das entsprechende Feld vor einer Eingabe am Bildschirm gelöscht werden soll. Die Standardvorgabe hierbei ist immer „J". Wenn Sie später eine Variable jedoch nur in der Maske ändern wollen, so antworten Sie dem Rechner mit „N", denn dann bleibt die Variable auf dem Bildschirm sichtbar und kann beliebig geändert werden.

Bei der Frage nach dem akustischen Signal ist die Standardoption einer neuen Maske „N", bei einer zu editierenden Maske wird die Option der alten Maske ausgegeben. Wenn Sie also einen Piepston bei der Feldeingabe wünschen, so antworten Sie mit „J".

Als nächstes kommt die Variablenvereinbarung an die Reihe. Sie haben die Auswahl zwischen 5 Möglichkeiten:

Auswahl 0 bedeutet keine Vereinbarung. Es entfällt bei diesem Feld also das SIZE Format.

Auswahl 1 läßt bei Eingabe nur Großbuchstaben zu.

Auswahl 2 läßt bei der Eingabe nur Zahlen zu.

Auswahl 3 läßt bei der Eingabe nur Zahlen sowie die Zeichen + – . E zu.

Bei Auswahl 4 können Sie selbst wählen, welche Zeichen und Zahlen bei der Eingabe zugelassen werden. Wenn Sie eine Maske ändern, so werden die Optionen der alten Maske ausgegeben, können jedoch beliebig geändert werden.

Der Rechner läuft alle Felder durch und fragt nach den Variablenvereinbarungen. Danach wird die generierte Maske ausgegeben. Sie sehen den Text und die USING Formate an den entsprechenden Bildschirmstellen.

Wollen Sie die Maske jetzt noch ändern, so drücken Sie die Funktionstaste BACK (FCTN 9), und Sie kommen wieder in den Leerbildschirm, allerdings wird die Maske bereits auf dem Bildschirm ausgegeben, und der Leerbildschirm hat jetzt schwarze Randstreifen. Sind Sie mit der Maske zufrieden, so drücken Sie die Funktionstaste PROC'D (FCTN 6). Sie werden nun gefragt, wo die Maske gespeichert werden soll. Geben Sie das entsprechende Gerät ein (z. B. CS1 oder DSK1.MASKE), und der Rechner speichert die Maske ab. Der Speichervorgang dauert je nach Anzahl der Felder mit dem Cassettenrecorder ca. 4 bis 8 Minuten, mit einem Diskettenlaufwerk ca. 8 bis 12 Sekunden.

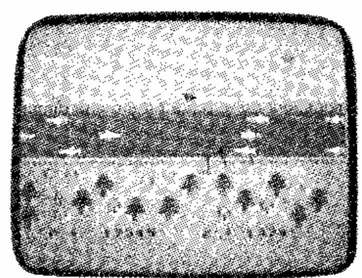
Lassen Sie sich von der Anleitung nicht beirren, denn die Maskenerstellung ist relativ einfach. Es hört sich alles viel schwieriger an, als es in Wirklichkeit ist. Probieren Sie es einfach einmal aus, und denken Sie daran, daß ein Programm niemals den Computer kaputtmachen kann. Es kann höchstens zum Systemabsturz kommen, was sich jedoch mit Ausschalten ohne weiteres wieder beheben läßt.




```
100 !@P-
110 REM ********************
120 REM *  SCREEN  PRINTER  *
130 REM *------------------*
140 REM *  fuer TI 99 / 4A  *
150 REM *  + Extended BASIC *
160 REM *  + ext. Speicher  *
170 REM *------------------*
180 REM * (c)1984 Tronicsoft*
190 REM *         V.Becker  *
200 REM ********************
210 REM
220 REM
230 !@P+
240 DIM MASKE$(24),VALD$(25),SZE(25),DAT(25),USER$(25),TON(25)
250 GOTO 270 :: DATEI$ :: DATUM$ :: DRUCK$ :: I :: K :: P :: W1 :: W2 :: CALL CL
EAR :: CALL GCHAR :: CALL SCREEN
260 !@P-
270 CALL SCREEN(4)
280 DISPLAY AT(1,1)ERASE ALL:"S C R E E N    P R I N T E RAufschluesselung einer
 durch SCREEN DESIGNER erstellten        Bilschirmmaske        ----------------
----------"
290 DISPLAY AT(7,1):"Name der aufzuschluesselndenMaske : (Geraet.name)"
300 ACCEPT AT(9,1)BEEP:DATEI$
310 ON ERROR 300
320 OPEN £1:DATEI$,INPUT ,SEQUENTIAL,INTERNAL,FIXED 80
330 FOR I=1 TO 24 :: INPUT £1:MASKE$(I):: NEXT I
340 FOR I=1 TO 25 :: INPUT £1:SZE(I),TON(I),DAT(I),VALD$(I),USER$(I)
350 IF DAT(I)=0 THEN 370
360 NEXT I
370 CLOSE £1
380 ON ERROR STOP
390 DISPLAY AT(12,1):"Drucker : " :: ACCEPT AT(12,11)BEEP:DRUCK$
400 ON ERROR 390
410 OPEN £1:DRUCK$
420 ON ERROR STOP
430 DISPLAY AT(15,1)BEEP:"Heutiges Datum : " :: ACCEPT AT(15,18)VALIDATE(NUMERIC
):DATUM$
440 PRINT £1:"     Aufschluesselung der Maske : ";DATEI$;"     ";DATUM$
450 FOR I=1 TO 80 :: PRINT £1:"-";:: NEXT I :: PRINT £1
460 CALL CLEAR
470 FOR I=1 TO 24 :: DISPLAY AT(I,1):MASKE$(I):: NEXT I
480 FOR I=1 TO 25
490 IF DAT(I)=0 THEN 530
500 W1=(DAT(I)-INT(DAT(I)))*100 :: W2=INT(DAT(I))
510 DISPLAY AT(W2,W1)SIZE(SZE(I)):USER$(I)
520 NEXT I
530 PRINT £1 :: PRINT £1 :: PRINT £1:CHR$(14)
540 PRINT £1:"      1234567890123456789012345678"
550 PRINT £1:"      ------------------------------"
560 FOR I=1 TO 24
570 PRINT £1:"   ";:: PRINT £1,USING "££":I;:: PRINT £1:":";
580 FOR K=3 TO 30 :: CALL GCHAR(I,K,P):: PRINT £1:CHR$(P);:: NEXT K
590 PRINT £1:":"
600 NEXT I
610 PRINT £1:"      ------------------------------"
620 PRINT £1:CHR$(15)! Umschaltung auf Normalschrift : Je nach Drucker bitte aba
endern
630 PRINT £1
640 PRINT £1:"    Feld Nr. : Zeile : Spalte : Laenge : Akkustisches Signal : Var
iablen Typ"
650 FOR I=1 TO 80 :: PRINT £1:"-";:: NEXT I :: PRINT £1
```

```
660 FOR I=1 TO 25
670 IF DAT(I)=0 THEN 800
680 PRINT £1:"        ";:: PRINT £1,USING "££":I;:: PRINT £1:"    :    ";
690 PRINT £1,USING "££":INT(DAT(I));:: PRINT £1:"   :    ";:: PRINT £1,USING "££":
(DAT(I)-INT(DAT(I)))*100;
700 PRINT £1:"   :   ";:: PRINT £1,USING "£££":SZE(I);:: PRINT £1:"    :         "
;
710 IF TON(I)=1 THEN PRINT £1,USING " ££ ":"Ja";ELSE PRINT £1,USING "££££":"nein
";
720 PRINT £1:"          : ";
730 IF LEN(VALD$(I))>1 THEN PRINT £1:VALD$(I):: GOTO 790
740 IF VALD$(I)="0" THEN PRINT £1:"keine Vereinb." :: GOTO 790
750 ON VAL(VALD$(I))GOTO 760,770,780
760 PRINT £1:"UALPHA" :: GOTO 790
770 PRINT £1:"DIGIT" :: GOTO 790
780 PRINT £1:"NUMERIC"
790 NEXT I
800 FOR I=1 TO 80 :: PRINT £1:"-";:: NEXT I
810 PRINT £1 :: PRINT £1
820 PRINT £1:"    Ende der Aufschluesselung"
830 CLOSE £1
840 CALL CLEAR
850 STOP
860 END
```

Mother Duck



Sind Sie schon einmal in die Rolle einer Entenmutter geschlüpft, die verzweifelt versucht, ihre Kinder zu füttern?

Wenn nicht, so können Sie es mit diesem Programm tun. Sie brauchen hierzu einen TI 99/4 A Homecomputer, das TI Extended BASIC Modul, einen Joystick und einen Cassettenrecorder, um das Programm abzuspeichern, denn Sie wollen ja sicherlich nicht vor jedem Spiel das komplette Programm eintippen.

Helfen Sie der Entenmutter, Ihre Kinder zu füttern

Tippen Sie das Programm ein und sichern Sie dieses dann auf Cassette.
Zum Spiel legen Sie nun die Cassette ein und starten mit RUN „CS1".
Kurze Zeit später erscheint auf dem Bildschirm das Titelbild mit der Frage: BEGINN BEI RUNDE ‹1–10›. Der Computer spielt Ihnen hierbei eine Melodie vor, bis Sie auf dem Keyboard irgend eine Taste drücken. Haben Sie jetzt eine Taste gedrückt, so erscheint der blinkende Cursor und Sie können die Runde eingeben, bei welcher Sie beginnen wollen.
Geben Sie eine Runde größer 10 ein, so beginnt das Spiel bei Runde 10.

Haben Sie die Runde eingegeben, so baut der Rechner das Spielfeld auf. Dies kann jedoch bei steigender Rundenzahl ein klein wenig dauern, da das Programm rein in BASIC geschrieben ist und beim Aufbau des unteren Bildschirmteiles (Wald) eine Menge Bildschirmvergleiche durchgeführt werden müssen.
Wenn das Bild nun generiert ist, so erscheint in der Mitte des oberen Flußufers unsere Entenmutter und der Computer spielt uns SALOME vor.
Um die Runde zu beginnen, drücken wir den roten Aktionsknopf am Joystick.

Ziel des Spieles

Die Ente muß über den Fluß in den Wald geführt werden, um dort die Würfel aufzusammeln.
Diese Würfel muß sie wieder über den Fluß auf die obere Seite des Bildschirmes bringen, um damit eine Pyramide aufbauen zu können.
Nur so kann unsere Entenmutter die Kirsche oben rechts im Bild erreichen.
Die Ente kann jedoch immer nur einen Würfel tragen. Hat unsere Ente nun die Kirsche erreicht, so muß sie diese zum Nest mit ihren Kindern bringen.
Das alles hört sich jedoch einfacher an, als es ist, denn es gibt viele Möglichkeiten, wie unsere Ente das Leben verlieren kann.
Wenn unsere Ente den Fluß überquert, so sollte sie sich am besten mit keinem Schwertfisch einlassen, denn dieser Schwertfisch könnte schlecht gelaunt sein, und unserer Ente den Garaus machen.
Die Ente bewegt sich im Wasser schneller als auf dem Land. Deshalb ist es ratsam, wenn die Ente den Fluß in Richtung Wald verläßt, dies nicht im Bereich eines Würfels zu tun, da es unserer Ente durch den Aufprall ein Leben kosten könnte.
Unsere Ente darf auf dem Landweg den Bildschirm nicht verlassen, da sie sonst stirbt.
Auch sollte die Ente nicht versuchen, durch einen Baum zu gehen, sie könnte sich verletzen und im Wald sterben.
Wenn unsere Ente endlich die Kirsche erreicht hat, so kommt aus dem Wald der böse Schrat und versucht, ihr die Kirsche abzunehmen.



Gelingt dem bösen Schrat dieses, so verliert unsere Ente ein Leben.
In dieser Spielphase darf die Ente auch nicht in den Fluß, da sie sonst ertrinkt. Steuern Sie die Ente zum Nest.

Steuerung

Die Ente wird mit dem Joystick in 8 Richtungen bewegt. Zum Aufnehmen eines Würfels gehen Sie mit der Ente über den Würfel und drücken Sie den Aktionsknopf. Wenn Sie mit dem Würfel die andere Seite des Flußufers erreicht haben, so wird der Würfel automatisch gutgeschrieben. Es ertönt ein Glockenschlag und der Würfel erscheint oben links im Bild.
Nachdem alle 10 Würfel gesammelt wurden, erscheint die Pyramide unter der Kirsche und die Ente sitzt ganz oben auf der Pyramide.
Drücken Sie jetzt den Aktionsknopf, um fortzufahren. Der Schrat erscheint und verfolgt die Ente. Führen Sie die Ente zum Nest mit den Kindern.
Wenn Sie das Nest erreicht haben, so ertönt wieder der Glockenschlag, und es geht mit der nächsten Runde weiter.
Jedesmal, wenn sie SALOME hören, müssen Sie den Aktionsknopf drücken, um die nächste Runde zu beginnen.

Punktwertung

Die Punktanzeige ist unten links im Bild eingeblendet und mit „P" gekennzeichnet.
Rechts daneben ist mit „Z" gekennzeichnet der Zeitbonus eingeblendet.
Dieser Zeitbonus hat zu Beginn jeder Runde den Wert 2000 und verringert sich im Laufe des Spieles.
Hat der Zeitbonus den Wert 0 erreicht, so stirbt die Entenmutter eines ihrer 5 Leben.
Wenn Sie die Kirsche erreicht haben, so erhalten Sie 1000 Punkte.
Haben Sie die Kirsche den Entenkindern gebracht, so wird der verbleibende Zeitbonus Ihrem Punktekonto gutgeschrieben.
Für jeden Würfel, den Sie gesammelt haben und auf die andere Seite des Flusses gebracht haben, erhalten Sie je nach Farbe:

weiß: 20 Punkte
blau: 40 Punkte
rot: 60 Punkte
grün: 80 Punkte

auf Ihrem Punktekonto gutgeschrieben.
In der oberen linken Ecke des Bildschirmes ist die Anzahl der Reserveenten angezeigt. Wenn alle 5 Enten gestorben sind, wechselt der Bildschirm die Farbe auf rot, und SALOME erklingt. Drücken Sie jetzt den roten Aktionsknopf, um ein neues Spiel zu beginnen.


```
1 REM  **************************
2 REM  * M o t h e r - D U C K  *
3 REM  *------------------------*
4 REM  *   TI 99/4 A            *
5 REM  * + TI Extended BASIC    *
6 REM  * + Joystick             *
7 REM  *------------------------*
8 REM  * (c) 1984 Volker Becker *
9 REM  * Steinbacher Stra~e 10  *
10 REM * 6370 Oberursel 6       *
11 REM **************************
12 REM
13 REM
14 REM
20 GOTO 80 :: A :: A1 :: A2 :: B1 :: B2 :: COLR :: D :: I :: K :: RUNDE :: S ::
SND :: ST :: SW :: T :: VGL :: WFARBE :: X :: Y :: Z :: ZW
30 CALL BILD :: CALL CHAR :: CALL CHARSET :: CALL CLEAR :: CALL COINC :: CALL CO
LOR :: CALL DELSPRITE :: CALL ENTE :: CALL GCHAR :: CALL GRAPH :: CALL HCHAR
40 CALL JOYST :: CALL KEY :: CALL LOCATE :: CALL MOTION :: CALL PATTERN :: CALL
POSITION :: CALL PUNKTE :: CALL SALOME :: CALL SCREEN :: CALL SOUND
50 CALL SPRITE :: CALL TITEL :: CALL WALD
80 PUNKT=0 :: ZEIT=2000 :: DUCK=4 :: WUERFEL=0 :: STATUS=0
89 OPTION BASE 1
90 DIM V(6),COL(10),FARBE(4),B(4)
91 DATA 16,5,7,13
92 !@P-
93 RESTORE 91
94 FOR I=1 TO 4 :: READ COLR :: FARBE(I)=COLR :: NEXT I
```

```
110 CALL CLEAR :: CALL CHARSET :: CALL SCREEN(4)
125 RANDOMIZE
130 CALL GRAPH
450 CALL TITEL(RUNDE,B())
455 CALL BILD
456 CALL CHAR(76,"03070507031F23272F2F275302020206C0E0A0E0C0F8C4E4F4F4E4CA404040
60")
460 CALL WALD(V(),FARBE(),RUNDE,COL())
462 CALL PUNKTE(ZEIT,PUNKT)
465 CALL SPRITE(#1,36,2,42,128)
470 CALL SALOME(B(),1)
800 CALL CHAR(68,"0000000000000000072F3F1F04060000000000000000307CF0C0E0E080C000
00")
810 CALL CHAR(72,"0000000000000073F0F000000000000000000000000000030FC30")
1000 CALL POSITION(#1,Z,S)
1005 ZEIT=ZEIT-2 :: CALL PUNKTE(ZEIT,PUNKT):: IF ZEIT<=0 THEN CALL ENTE(DUCK,ZW,
SW,WFARBE,I,FARBE(),STATUS):: ZEIT=2000
1007 IF DUCK<0 THEN 9000
1010 CALL JOYST(1,X,Y):: IF X=0 THEN 1100
1020 IF X=4 THEN CALL PATTERN(#1,68)ELSE CALL PATTERN(#1,36)
1100 IF X=0 AND Y=0 THEN 1110 ELSE CALL SOUND(-30,-6,0)
1110 IF Z<58 THEN 2000 ELSE 3000
2000 IF Y=4 AND Z<38 THEN Y=0
2005 IF STATUS=1 THEN 5000
2010 CALL MOTION(#1,-Y*1.5,X*1.5)
2020 GOTO 1000
3000 IF Z>106 THEN 4000
3005 CALL COINC(ALL,ST):: IF ST THEN CALL ENTE(DUCK,ZW,SW,WFARBE,I,FARBE(),STATU
S)
3010 CALL MOTION(#1,-Y*2,X*2):: GOTO 1000
4000 A=INT((Z+7)/8+1):: D=INT((S+7)/8+1):: IF A>24 OR D>32 OR D<1 THEN CALL ENTE
(DUCK,ZW,SW,WFARBE,I,FARBE(),STATUS):: GOTO 1000
4020 CALL MOTION(#1,-Y*1.5,X*1.5)
4040 CALL GCHAR(A,D,VGL):: IF (VGL>135 AND VGL<140)OR(VGL>39 AND VGL<42)THEN CAL
L ENTE(DUCK,ZW,SW,WFARBE,I,FARBE(),STATUS):: GOTO 1000
4060 CALL KEY(1,T,S):: IF T=18 THEN 4100 ELSE 1000
4100 CALL MOTION(#1,0,0)
4105 IF STATUS=1 THEN 1000
4110 FOR I=11 TO 20
4120 CALL COINC(#1,#I,12,ST)
4130 IF ST THEN 4200
4140 NEXT I
4150 GOTO 1000
4200 CALL POSITION(#I,ZW,SW):: WFARBE=COL(I-10):: CALL DELSPRITE(#I):: STATUS=1
4210 DATA 261,329,391,523,30000,523,391,329,261
4220 RESTORE 4210
4230 FOR K=1 TO 9 :: READ SND :: CALL SOUND(-250,SND,0):: NEXT K
4240 GOTO 1000
5000 WUERFEL=WUERFEL+1
5002 CALL MOTION(#1,0,0)
5005 STATUS=0
5010 CALL HCHAR(2,3,101,WUERFEL)
5012 CALL PUNKTE(ZEIT,PUNKT)
5013 FOR K=0 TO 30 :: CALL SOUND(-250,523,K,440,K):: NEXT K
5015 PUNKT=PUNKT+20*WFARBE
5020 IF WUERFEL=10 THEN WUERFEL=0 :: CALL HCHAR(2,3,128,10):: GOTO 6000
5030 GOTO 1000
6000 CALL COLOR(10,2,11,11,2,11):: CALL MOTION(#1,0,0):: CALL LOCATE(#1,4,205)
6005 RESTORE 4210
6007 FOR K=1 TO 9 :: READ SND :: CALL SOUND(-250,SND,0):: NEXT K
6010 PUNKT=PUNKT+1000 :: CALL PUNKTE(ZEIT,PUNKT)
6020 CALL KEY(1,T,S):: IF T=18 THEN 6030 ELSE 6020
```



```
6030 CALL DELSPRITE(#28):: CALL COLOR(10,11,11,11,11,11)
6040 CALL POSITION(#1,A2,B2):: CALL POSITION(#2,A1,B1):: A=INT((A2+7)/8+1):: D=I
NT((B2+7)/8+1)
6050 IF A<1 OR A>8 OR D<1 OR D>32 THEN 7000
6060 CALL GCHAR(A,D,VGL):: IF VGL=122 OR VGL=123 THEN 7500
6070 CALL MOTION(#2,(A2-A1)/4,(B2-B1)/4)
6080 CALL JOYST(1,X,Y):: CALL MOTION(#1,-Y*1.5,X*1.5):: CALL SOUND(-30,-6,0):: I
F X=4 THEN CALL PATTERN(#1,68)ELSE IF X=-4 THEN CALL PATTERN(#1,36)
6090 CALL COINC(#1,#2,10,ST):: IF ST THEN 7000
6100 ZEIT=ZEIT-2 :: CALL PUNKTE(ZEIT,PUNKT):: IF ZEIT<=0 THEN 7000 ELSE 6040
7000 CALL MOTION(#1,0,0,#2,0,0):: CALL ENTE(DUCK,ZW,SW,WFARBE,I,FARBE(),0)
7005 IF DUCK<0 THEN 9000
7010 CALL HCHAR(16,1,32,256):: GOTO 460
7500 CALL MOTION(#1,0,0,#2,0,0):: CALL DELSPRITE(#2)
7505 PUNKT=PUNKT+ZEIT :: ZEIT=0 :: CALL PUNKTE(ZEIT,PUNKT):: ZEIT=2000
7507 FOR K=0 TO 30 :: CALL SOUND(-250,523,K,440,K):: NEXT K
7510 RUNDE=RUNDE+1 :: IF RUNDE>10 THEN RUNDE=10
7520 CALL HCHAR(16,1,32,256)
7530 GOTO 460
8000 CALL ENTE(DUCK,ZW,SW,WFARBE,I,FARBE(),STATUS)
8010 GOTO 1000
9000 CALL DELSPRITE(ALL):: CALL SCREEN(7):: CALL SALOME(B(),1)
9010 GOTO 80
27999 !@P+
28000 SUB TADEL
28001 GOTO 28010 :: A :: L :: I :: CALL SOUND :: !@P-
28010 DATA 750,195,250,220,500,233,500,195,1000,277
28020 RESTORE 28010
28030 FOR I=1 TO 5 :: READ L,A :: CALL SOUND(L,A,0):: NEXT I
28035 !@P+
28040 SUBEND
28050 SUB PUNKTE(ZEIT,PUNKT)
28051 !@P-
28060 DISPLAY AT(24,1):USING "  P : ######     Z : ####":PUNKT,ZEIT
28070 !@P+
28080 SUBEND
28200 SUB ENTE(DUCK,ZW,SW,WFARBE,I,FARBE(),STATUS)
28201 GOTO 28210 :: CALL TADEL :: !@P-
28210 CALL MOTION(#1,0,0)
28220 CALL PATTERN(#1,72)
28230 CALL TADEL
28240 DUCK=DUCK-1
28245 CALL HCHAR(1,3,128,4)
28250 IF DUCK<=0 THEN 28265
28260 CALL HCHAR(1,3,121,DUCK)
28265 CALL SPRITE(#1,36,2,42,128)
28267 IF DUCK<0 THEN CALL DELSPRITE(#1)
28268 IF STATUS=0 THEN SUBEXIT
28269 CALL SPRITE(#I,96,FARBE(WFARBE),ZW,SW,0,0):: STATUS=0
28270 !@P+
28271 SUBEND
29000 SUB TITEL(RUNDE,B())
29001 GOTO 29010 :: Z$ :: I :: P :: CALL MAGNIFY :: CALL VCHAR :: !@P-
29010 CALL BILD
29020 DATA 98,D,114,U,130,C,146,K
29035 CALL MAGNIFY(2)
29040 DISPLAY AT(9,9)SIZE(12):"   MOTHER   " :: DISPLAY AT(14,9)SIZE(12):"
"
29050 CALL VCHAR(10,11,32,4):: CALL VCHAR(10,22,32,4)
29053 DISPLAY AT(24,3):"@ 1984 BY VOLKER BECKER"
29055 RESTORE 29020
29060 FOR I=1 TO 4 :: READ P,Z$ :: CALL SPRITE(#I,ASC(Z$),11,81,P):: NEXT I
```

```
29070 DISPLAY AT(18,1):"BEGINN BEI RUNDE <1-10> :" :: CALL SALOME(B(),5)
29080 ACCEPT AT(18,27)SIZE(2)VALIDATE(DIGIT):RUNDE
29090 IF RUNDE<1 THEN RUNDE=1
29100 IF RUNDE>10 THEN RUNDE=10
29110 CALL HCHAR(16,1,32,288):: CALL DELSPRITE(ALL)
29115 !@P+
29120 SUBEND
30000 SUB SALOME(B(),KY)
30001 GOTO 30010 :: AZ :: I :: J :: K :: M1 :: M2 :: S :: STV :: T :: !@P-
30010 DATA 2,220,261,349,440,6,261,440,2,293,466
30020 DATA 4,195,233,293,391,6,233,391,1,349,440,1,391,466,1,329,391,2,349,440,1
,391,466,2,466,587,2,391,466
30040 DATA 2,220,261,349,440,8,440,698
30045 DATA 2,220,261,329,440,6,523,659,2,329,523
30050 DATA 1,195,233,293,391,4,391,466
30055 DATA 1,233,261,329,440,2,391,466,1,329,523,1,261,440,1,195,233,293,391,1,3
29,391,2,391,440,1,329,391
30060 DATA 1,195,233,293,391,2,233,293,2,261,329
30065 DATA 2,220,261,349,440,7,261,440,1,30000,30000
30070 DATA 2,220,261,349,440,6,261,440,2,293,466
30075 DATA 4,195,233,293,391,6,233,391,1,349,440,1,391,466,1,329,391,2,349,440,1
,391,466,2,466,587,2,391,466
30085 DATA 2,220,261,349,440,7,440,698,1,30000,30000
30087 DATA 4,220,261,349,440,6,440,698,2,391,659,6,329,523,1,391,466,1,440,523
30091 DATA 2,195,233,293,391,4,440,554,2,440,523,2,261,440
30093 DATA 2,220,261,349,440,8,220,349,0
30100 RESTORE 30010
30105 AZ,STV=0
30110 READ K :: IF K=0 THEN 30100
30115 READ B(1),B(2),B(3),B(4)
30120 FOR I=1 TO K
30130 FOR J=1 TO 4
30135 STV=STV+1 :: IF STV>AZ THEN READ AZ,M1,M2 :: STV=1
30140 CALL SOUND(400,M1,5,M2,4,B(J),9):: CALL KEY(KY,T,S):: IF S<>0 THEN SUBEXIT
30150 NEXT J :: NEXT I
30160 GOTO 30110
30170 !@P+
30200 SUBEND
31000 SUB BILD
31100 GOTO 31165 :: CALL PYRAMIDE :: !@P-
31165 CALL COLOR(13,1,11):: CALL HCHAR(1,1,128,224)
31170 CALL COLOR(10,11,11,11,11,11,14,13,1,2,7,1):: CALL PYRAMIDE
31180 CALL COLOR(10,2,11,11,2,11)
31190 CALL COLOR(9,5,11,12,7,11):: CALL HCHAR(9,1,100,192):: CALL HCHAR(8,1,120,
32):: CALL HCHAR(15,1,120,32)
31195 CALL HCHAR(1,3,121,4)
31197 CALL HCHAR(7,5,122):: CALL HCHAR(7,6,123)
31199 !@P+
31200 SUBEND
31210 SUB WALD(V(),FARBE(),RUNDE,COL())
31211 GOTO 31212 :: C :: I :: K :: SP :: Z :: !@P-
31212 CALL MAGNIFY(3)
31215 FOR I=1 TO RUNDE+2
31220 Z=RND*5+16 :: SP=RND*30+1 :: CALL GCHAR(Z,SP,V(1)):: CALL GCHAR(Z+1,SP,V(2
)):: CALL GCHAR(Z,SP+1,V(3)):: CALL GCHAR(Z+1,SP+1,V(4))
31222 CALL GCHAR(Z+2,SP,V(5)):: CALL GCHAR(Z+2,SP+1,V(6))
31223 FOR K=1 TO 6 :: IF V(K)<>32 THEN 31220
31224 NEXT K
31225 CALL HCHAR(Z,SP,136):: CALL HCHAR(Z+1,SP,137):: CALL HCHAR(Z,SP+1,138):: C
ALL HCHAR(Z+1,SP+1,139)
31230 CALL HCHAR(Z+2,SP,40):: CALL HCHAR(Z+2,SP+1,41):: NEXT I
31315 FOR I=11 TO 20
```

```
31320 Z=INT(RND*6+16):: SP=INT(RND*29+1):: CALL GCHAR(Z,SP,V(1)):: CALL GCHAR(Z+
1,SP,V(2)):: CALL GCHAR(Z,SP+1,V(3)):: CALL GCHAR(Z+1,SP+1,V(4))
31322 CALL GCHAR(Z+2,SP,V(5)):: CALL GCHAR(Z+2,SP+1,V(6))
31323 FOR K=1 TO 6 :: IF V(K)<>32 THEN 31320
31324 NEXT K
31325 C=INT(RND*3+1):: COL(I-10)=C :: CALL SPRITE(#I,96,FARBE(C),Z*8-7,SP*8-7)
31327 CALL HCHAR(Z,SP,33,2):: CALL HCHAR(Z+1,SP,33,2)
31330 NEXT I
31340 CALL SPRITE(#28,124,7,1,205)
31345 CALL COLOR(10,11,11,11,11,11)
31350 CALL SPRITE(#3,140,11,65,240):: CALL SPRITE(#4,132,11,81,5):: CALL SPRITE(
#5,140,11,97,240)
31360 IF RUNDE>1 THEN CALL SPRITE(#6,132,11,81,70)
31370 IF RUNDE>2 THEN CALL SPRITE(#7,140,11,65,180)
31380 IF RUNDE>3 THEN CALL SPRITE(#8,140,11,97,170)
31390 IF RUNDE>5 THEN CALL SPRITE(#9,132,11,81,175)
31400 IF RUNDE>7 THEN CALL SPRITE(#21,140,11,97,40)
31410 IF RUNDE>=10 THEN CALL SPRITE(#22,140,11,65,30)
31420 CALL MOTION(#3,0,-(8+RUNDE),#7,0,-(8+RUNDE),#22,0,-(8+RUNDE))
31430 CALL MOTION(#4,0,4+RUNDE,#6,0,4+RUNDE,#9,0,4+RUNDE)
31440 CALL MOTION(#5,0,-(1+RUNDE),#8,0,-(1+RUNDE),#21,0,-(1+RUNDE))
31445 CALL SPRITE(#2,76,14,193,8,0,0)
31447 !@P+
31450 SUBEND
32000 SUB PYRAMIDE
32001 !@P-
32010 DISPLAY AT(3,24)SIZE(2):"hi" :: DISPLAY AT(4,24)SIZE(3):"jkl" :: DISPLAY A
T(5,23)SIZE(4):"mnno" :: DISPLAY AT(6,23)SIZE(5):"pqqrs"
32020 DISPLAY AT(7,23)SIZE(5):"ttttu"
32025 !@P+
32030 SUBEND
32040 SUB GRAPH
32045 !@P-
32080 CALL CHAR(100,RPT$("F",16)):: CALL CHAR(120,"66FFF9FD9FBFFF99")
32097 CALL CHAR(33,RPT$("0",16))
32100 CALL CHAR(96,"000000010204080C0A0908080402010000000804020100818284888890A0
C080")
32102 CALL CHAR(136,"0103070F1F03070F1F3F7F0F1F3F7FFF80C0E0F0F8C0E0F0F8FCFEF0F8F
CFEFF"):: CALL CHAR(40,RPT$("03",8)):: CALL CHAR(41,RPT$("C0",8))
32105 CALL CHAR(128,RPT$("0",16))
32130 CALL CHAR(104,"00010204080C0A098040201008182848080814224180C1A288894A2C18
0C1A2")
32135 CALL CHAR(124,"0000000001F3F7F7F7F7F7F3F3F1F0000000608102040 80E0F0F0F0E
0C000")
32140 CALL CHAR(108,"000000000080808000000001020408 0C948888492A1C081C808080402 01
00818")
32150 CALL CHAR(112,"0A0908081422418 02A49888894A2C18028488888 94A2C1800000000000
00080")
32155 CALL CHAR(140,"0000000000000003FDFF070300000000000000002060E1F3FFFEFFF3C18
00000")
32156 CALL CHAR(132,"000000000040687CFFF7FFFCF8302000000000000000000C0BFFFE0C0000
00000")
32160 CALL CHAR(116,"C1A2948888492A1C8080808080000000 08")
32170 CALL CHAR(36,"000000000000C3E0F0307070103000000000000000000000E0F4FCF82060
0000")
32180 CALL CHAR(121,"00060FBCFC101800")
32190 CALL CHAR(101,"1028446C54543810")
32200 CALL CHAR(122,"617942FF341B0E078CEF10FFBC5870E0")
32210 CALL CHAR(64,"3C4299A1A199423C")
32220 !@P+
32300 SUBEND
```

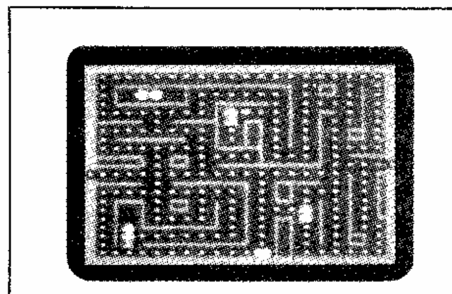


Topprogramm

Fressmann ist natürlich eine Pacman-Version. Unser hier abgedrucktes Programm, welches ohne Programmerweiterung lauffähig ist, steht der Originalversion jedoch in nichts nach.

Als Topprogramm ausgewählt von der Redaktion.

Der Spieler steuert seinen Fressmann mit Joystick. Fressmann wird von drei Gespenstern gejagd. Fette Punkte, die im Labyrinth herumliegen, dienen Fressmann wie Gespenstern gleichermaßen zur Nahrung. Nach dem Verzehr eines – fetten Punktes – ist Fressmann kurzfristig in der Lage, ein Gespenst zu verschlucken. Ab und zu tauchen Früchte auf, bei deren Verzehr ein fetter Punkt neu ins Feld gelegt wird. Am Ende des Spieles wird der Score und während des Spieles nur die Bonuspunkte ausgedruckt (aus Geschwindigkeits- und Speichergründen).



Programm Teil 1

```
10000 DATA24,24,24,24,24,24,24,24,0,0,0,255,255,0,0,0,0,0,0,31,31,24,24,24
10010 DATA0,0,0,248,248,24,24,24,24,24,24,248,248,0,0,0,24,24,24,31,31,0,0,0
10020 DATA24,24,24,31,31,24,24,24,24,24,24,248,248,24,24,24,24,24,24,255,255,0,0
,0
10030 DATA0,0,0,255,255,24,24,24,0,0,0,0,24,24,24,24
10040 DATA24,24,24,24,0,0,0,0,0,0,0,240,240,0,0,0,0,0,0,15,15,0,0,0
10050 DATA255,255,255,255,255,255,255,255,56,124,182,146,254,254,254,170
10060 DATA0,0,0,24,24,0,0,0,24,36,96,80,72,76,204,192
10070 DATA0,6,25,120,248,240,240,96,14,17,124,254,254,254,124,56
10075 DATA0,0,60,60,60,0,0,0
10080 DATA56,68,202,238,130,130,254,170,255,231,231,129,129,231,231,255
10085 DATA0,0,0,0,0,0,0,0
10100 DATA56,124,254,252,248,254,124,56,28,62,127,63,31,127,62,28
10110 DATA36,110,255,255,255,255,126,60,60,126,255,255,255,255,110,36
10150 DATA56,124,254,254,254,254,108,36,0,56,124,254,254,254,236,100
10160 DATA0,0,56,124,254,254,254,236,0,0,0,56,124,254,254,254,0,0,0,0,56,124,254
,254
10170 DATA0,0,0,0,0,56,124,254,0,0,0,0,0,0,56,124,0,0,0,0,0,0,0,56
20000 FORT=7168TO7351:READB:POKET,B:NEXT
20100 FORT=7424TO7527:READB:POKET,B:NEXT
20200 FORT=7528TO7551:POKET,0:NEXT
20300 FORT=0TO79:S=PEEK(33152+T):POKE7552+T,255-(SORS/2):NEXT
```

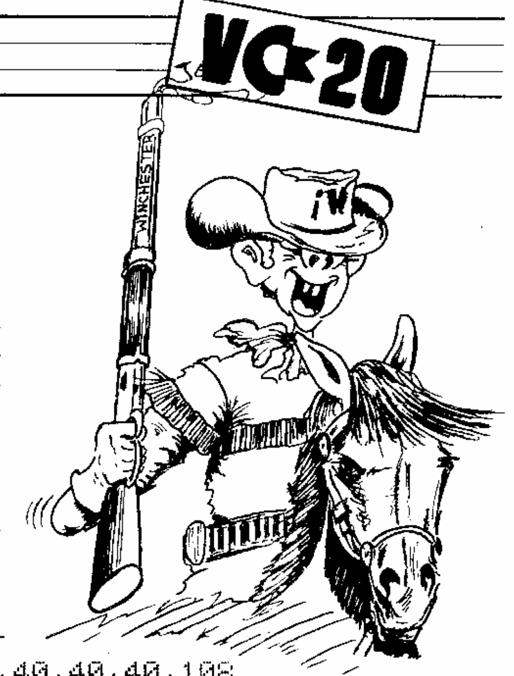
Programm Teil 2

```
20 POKE36878,15:POKE36879,8:POKE36869,255:PRINT"[illegible]NNNNNNNNNNNNNNNNNNNNN";
22 PRINT"NPPPPPPPPPPPPPPPPPPPPPNNPBAAAAIAAAAAACPBCPJPNNP@[illegible] !! [illegible]@PPPPPP@PEDP@PN";
28 PRINT"NPEAAAADPBAAAP@PPPP@PNNPPPPPPPPP@PPPP@PBLP@PNNAAALPMAAGPBCP@P@PP@PN";
34 PRINT"NPPPPPPPP@P@@P@P@PBGPNNPMAAAALP@P@@P@P@PEDPNNPPPPPPPPPEAHGPKP@PPPPN";
40 PRINT"NAAACPBCPPPP@PPP@PMAANNAAADPEDPMAAHAAADPPPPPPPPPPPPPPPPPPPPPPJPBN";
46 PRINT"NAAALPMAAAACPBIALP@P@NNPPPPPPPPPPP@P@@PPP@P@NNPBACPBCPJP@PEDPJP@P@N";
52 PRINT"NP@P@PEDP@P@PPPP@P@P@NNP@P@PPPP@P@PBCP@P@PENNP@PEAAAADP@P@@P@P@PPN";
58 PRINT"NP@PPPPPPPP@P@@P@PFCPNNPKPMAAAAAAIP@@PKPEDPNNPPPPPPPPPPPPP@@PPPPPPN";
```



```
64 PRINT"NNNNNNNNNNNNNNNNNNNNNN";:POKE8185,14:POKE38905,6:POKE37154,127
67 ZX=INT(RND(1)*460)+7703:IFPEEK(ZX)<>16THEN67
70 FORT=7703TO8163:IFPEEK(T)=16THENPOKET+30720,7
120 NEXT:P=3:A$(1)="V1000":A$(2)="V1500":A$(3)="V2000":A$(4)="V3000":G=36876:POK
EZX,20
130 G1=8035:G2=7866:Y2(1)=1:Y2(2)=-1:Y2(3)=22:Y2(4)=-22:U1=4:G3=8003:Y3(1)=1:Y3(
2)=-1
140 Y3(3)=22:Y3(4)=-22:U2=3:A=8154:C=35:BN=15:POKE38586,5
200 FORQ=1TO78:READX,Y1:FORO=1TOX:POKEG1,32:G1=G1+Y1:POKEG1,BN:POKEG2,16:G2=G2+Y
2(U1)
320 IFU1<3THENGOTO700
330 IFPEEK(G2)<15THENG2=G2-Y2(U1):U1=INT(RND(1)*4)+1
340 POKEG2,BN:POKEG3,16:G3=G3+Y3(U2):IFPEEK(G3)<15THENG3=G3-Y3(U2):U2=INT(RND(1)
*4)+1
360 POKEG3,BN:IFG1=AORG2=AORG3=ATHEN4000
405 T=PEEK(37151):IFPEEK(37152)=119THENPOKEA,32:A=A+1:B=-1:C=33:GOTO430
410 IFT=110THENPOKEA,32:A=A-1:B=1:C=34:GOTO430
415 IFT=118THENPOKEA,32:A=A+22:B=-22:C=36:GOTO430
420 IFT=122THENPOKEA,32:A=A-22:B=22:C=35
430 IFPEEK(A)<32THEN800
440 POKEA,C:NEXTO,Q:RESTORE:GOTO200
700 IFPEEK(G2+22)>15THENU1=3:BN=15:GOTO330
710 IFPEEK(G2-22)>15THENU1=4:GOTO750
730 GOTO330
750 IFS=0THENIFRND(1)*10>7THENPOKE7866,INT(RND(1)*3)+17:S=1:GOTO3000
760 IFSTHENS=S+1:IFS=4THENPOKE7866,32:S=0
770 GOTO330
800 IFPEEK(A)>14THEN900
820 A=A+B:GOTO440
900 IFPEEK(A)=16THENPOKEG,245:E=E+10:POKEG,0:GOTO440
960 IFPEEK(A)=17THENE=E+1000:I=1:GOTO2000
965 IFPEEK(A)=18THENE=E+1500:I=2:GOTO2000
970 IFPEEK(A)=19THENE=E+2000:I=3:GOTO2000
980 IFPEEK(A)=20THENPOKEG,240:E=E+200:BN=21:POKEG,0:GOTO440
982 IFPEEK(A)=21THENE=E+3000:I=4:GOTO2000
985 IFPEEK(A)=15THEN4000
990 POKEG,0:GOTO440
2000 POKEA,34:FORO5=1TO8:PRINT"■■■■■■■■"A$(I):POKEG1,15:POKEG2,15:POKEG3,15
2040 FORO1=200TO250STEP8:POKEG,O1:FORT=1TO13:NEXTT,O1:PRINT"■■■■■■■■■NNNNN"
2055 POKEG1,21:POKEG2,21:POKEG3,21:FORO2=250TO200STEP-8:POKEG,O2:FORT=1TO13:NEXT
T,O2,O5
2060 IFI=4THENA=8154:BN=15:GOTO2100
2070 POKE7911,20
2100 POKEG,0:GOTO440
3000 POKEG,230:FORF=1TO15:NEXT:POKEG,220:FORF=1TO15:NEXT:POKEG,240:FORF=1TO20:NE
XT
3020 POKEG,0:GOTO760
4000 IFBN=21THENE=E+3000:I=4:GOTO2000
4020 P=P-1:S=0:FORM=37TO45:POKEA,M:POKEG,180-(M-37)*5:FORF=1TO150:NEXT:POKEA,M+2
4050 FORF=1TO150:NEXTF,M:POKEG,0:IFP=0THENPRINT"■■■■■■■■"E"■ ":WAIT197,64,255:R
UN
4120 RESTORE:POKEG1,32:POKEG2,32:POKEG3,32:POKE7749+P,32:GOTO130
30000 DATA5,22,9,1,9,-22,5,1,9,22,3,1,3,-22,1,-1,7,-22,2,-1,5,-22,1,1,2,-22,3,-1
,6,22
30010 DATA2,-1,5,-22,3,-1,3,22,3,-22,3,1,2,-22,5,-1,2,22,7,-1,4,-22,14,1,9,22,9,
-22
30020 DATA3,1,5,22,5,-22,2,1,8,22,3,-1,12,22,2,-1,5,-22,3,-1,3,1,2,-22,2,1,3,-22
,10,1
30030 DATA5,22,3,1,3,-22,2,1,5,22,7,-1,2,22,2,-1,7,-22,9,1,5,-1,5,-22,4,-1,2,-22
,4,1
30040 DATA2,-22,2,22,3,1,2,22,7,-1,7,1,1,22,3,1,3,-1,2,22,9,1,12,-1,5,22,3,1
30050 DATA3,-22,2,1,5,22,7,-1,3,-22
```

Outlaw



für VC-20 ohne Speichererweiterung

Ein Action-Spiel für Wildwest-Fans. Spielen Sie gegen den Computer.

Nach dem Vorspann muß der Spielercowboy – also Ihre Figur – versuchen den Computercowboy zu treffen. Felsen und Kakteen dienen beiden als Deckung. Wird eine Figur getroffen, wird sie bewegungsunfähig (Schuß ins Bein) oder sie wird ganz vernichtet. Als Spieler stehen Ihnen 6 Schuß Munition im Revolver zur Verfügung. Ist der Revolver leergeschossen, muß er bei der Munitionskiste nachgeladen werden. Dies geschieht durch Druck auf den Feuerknopf. Bonuspunkte gibt es für eine getroffene Spielfigur. Bei erfolgreichem Spiel für alle 4 getroffenen Spielfiguren. Bei Outlaw muß zuerst *„Outlaw Res."* geladen werden. Nach Ablauf des Programmes, welches die Zeichendatas enthält, wird Outlaw einfach mit „load" nachgeladen und gestartet. Gesteuert wird mit Joystick.

```
1010 DATA56,254,56,16,254,186,186,186,186,124,108,40,40,40,40,108
1020 DATA0,0,0,0,56,254,56,16,56,59,62,56,124,62,226,131
1030 DATA0,0,0,0,28,127,28,8,28,220,124,28,62,124,71,193
1040 DATA24,27,219,219,223,248,24,24,24,27,219,219,251,31,24,24,0,8,26,26,14,24,
8,24
1050 DATA1,7,15,31,63,63,127,127,127,127,127,63,15,7,3,0
1060 DATA224,224,224,224,192,192,128,128,128,192,192,192,192,224,224,224
1070 DATA127,255,191,177,152,152,49,49,192,192,128,128,192,192,128,128
1080 DATA0,0,8,28,62,118,224,224,0,0,0,0,0,0,1,63
1090 DATA254,255,253,141,25,25,140,140,3,3,1,1,3,3,1,1
1100 DATA0,0,16,56,124,110,7,7,0,0,0,0,0,0,128,252
1110 DATA0,128,8,28,190,118,224,224,14,63,14,4,15,14,14,63
1120 DATA0,1,16,56,125,110,7,7,112,252,112,32,240,112,112,252
1130 DATA186,124,108,40,40,44,32,96,186,124,108,40,40,104,8,12
1140 DATA56,254,56,18,251,185,191,184,184,124,108,40,40,40,40,60
1150 DATA28,127,28,72,223,157,253,29,29,62,54,20,20,20,20,60
1160 DATA219,195,126,36,108,108,108,36,0,0,0,0,0,0,0,126,195,219,219,90,90,90,
219
1162 DATA24,24,126,126,24,24,24,24
1170 DATA240,144,135,244,54,182,247,0,0,0,189,37,53,53,189,0,0,0,222,18,158,152,
158,0
1180 DATA120,72,72,123,122,107,107,0,0,0,0,247,84,86,86,0,0,0,0,224,160,172,172,
0
1190 DATA0,242,146,247,255,247,146,242,0,0,17,31,16,31,1,0
1200 DATA0,0,15,31,63,31,15,0,0,0,240,248,252,248,240,0
1210 DATA0,255,0,0,0,0,0,0,144,34,85,40,44,58,127,255,9,68,170,20,52,90,254,255
1220 DATA62,34,38,58,50,50,62,0
1230 DATA28,4,4,12,12,12,30,0,62,34,2,62,48,48,62,0,30,2,2,30,6,38,62,0
1240 DATA32,32,36,36,62,12,12,0,30,16,16,30,6,38,62,0,62,32,32,62,50,50,62,0
1250 DATA62,34,4,8,24,24,24,0,30,18,18,62,50,50,62,0,62,34,34,62,6,6,62,0
1260 DATA124,68,254,68,68,254,68,124
3000 FORA=7168TO7639:READB:POKEA,B:NEXT
```

```
10 POKE36879,8:PRINT"            ** OUTLAW **                      "
15 PRINT"                    "SPC(11)"                                "
20 PRINT"    (C) 1983 BY M. KRAUS":POKE198,0:WAIT198,1
100 PRINT"      #$% 0    &'( 60 @@":POKE36869,255:POKE36879,127:POKE38905,0:POKE7
701,0
120 POKE38421,0:FORA=1TO20:PRINT"                              ";:NEXT:W=30720:POKE368
78,15
160 T1=36876:T2=T1-1:T3=T1+1:FORA=1TORND(1)*3+4:B=7726+INT(RND(1)*18)*22+RND(1)*
17
165 IFPEEK(B)<>32ORPEEK(B+1)<>32ORPEEK(B+23)<>32ORPEEK(B+22)<>32THENA=A-1:GOTO17
5
170 POKEB,9:POKEB+22,10:POKEB+23,11:POKEB+1,12
175 NEXT:FORA=2TO20:FORB=1TO3:C=7682+A*22+RND(1)*18
```

```
195 IFPEEK(C)<>32ORPEEK(C+1)<>32ORPEEK(C-1)<>32ORPEEK(C-44)<>32THENB=B-1:GOTO205
200 POKEC,RND(1)*2+6:POKEC+W,5
205 NEXTB,A:FORT=1TO500:NEXT:PRINT"■ VU■■■MN]"TAB(19)"WX■■■RQ]]":FORT=1TO1000:NE
XT:A=19
260 FORB=1TO7:A=A-1:PRINTTAB(B)" VU■■■■ MN]"TAB(A)"WX ■■■■RQ ]]":FORT=1TO150:NEX
TT,B
270 FORT=1TO500:NEXT:PRINTTAB(8)"PO@@ST■■■■■■■■MNAARQ]";:FORT=1TO500:NEXT:A=8:PRI
NT"]"
300 FORB=12TO18:A=A-1:PRINTTAB(B)" PO■■■■ MN]]":PRINTTAB(A)"ST ■■■■RQ ]]"
340 FORT=1TO150:NEXTT,B:PRINT"                [ ]                £ ↑         ]]";
350 FORT=1TO700:NEXT:A=10:PRINT"]":FORB=11TO19:A=A-1
360 PRINTTAB(A)"] ■■■↑ ]]":PRINTTAB(B)" [■■■ £]]":FORT=1TO200:NEXTT,B
390 PRINT"@ ■■■A ]";:POKE8184,32:POKE8162,32:POKE8185,1:POKE8163,0:ZX=.8
395 PRINT"■■■■  ■■■  ■■■ ■■■  ■■■    ■ ■■ :■ ■■■  ■■■ ■■■    ■■■ ■ ";
450 POKEA,32:POKEA+22,32:B=8163:A=8142:F=22:G=0:H=22:AM=60
500 G=F/2+11:F=-F
510 X=RND(1)*12:IFB+X*F<7702ORB+X*F>8163THEN500
520 IFX<5THENH=22:F=22:G=0
550 FORY=1TOX:POKEB+G,32:B=B+F:POKEB,0:POKEB+22,1
585 IFPEEK(B+1)<>32ANDRND(1)>ZXTHEN1000
615 POKET1,240:T=PEEK(37151):IFT=118ANDA<8141THENPOKET2,240:POKEA,32:A=A+H
620 IFT=122ANDA>7702THENPOKET2,240:POKEA+22,32:A=A-H
625 IFT=94THEN1500
630 POKEA,0:POKEA+22,1:POKET1,0:POKET2,0:NEXTY:GOTO500
700 PRINT"■■■■■■■■"SC"■"SPC(15)AM"■ ":RETURN
1000 POKET2,128:M=B+20:N=B+1:POKEB,4:POKEB+22,5:Z=1:POKET2,0
1010 FORK=MTONSTEP-1:IFPEEK(K)<>32THEN3000
1030 POKEK,197:POKEK+1,32:NEXT
1500 Q=Q+1:IFQ>6ANDPEEK(A+24)<>58THEN630
1505 POKET2,128:POKET1,0:M=A+24:N=A+43:Z=-1:POKEA,2:POKEA+22,3:POKET2,0
1510 FORK=MTON:IFPEEK(K)<>32THEN3000
1530 POKEK,197:POKEK-1,32:NEXT:POKEK-1,32:POKET3,0:GOTO630
3000 POKEK+Z,32:V=PEEK(K):IFV=58THENQ=0:AM=AM-6:GOSUB700:IFAM<0THEN5200
3010 U=PEEK(K+W):POKET1,0:POKET2,0:IFV>8THENPOKET3,0:GOTO630
3030 POKEK+W,2:FORT=1TO30:POKET1+2,15-T/2:POKEK,46:POKET3,180-T:POKEK,47:NEXT:PO
KEK+W,U
3050 IFV=8THENPOKEK,32:POKEK+W,6:IFZ=-1THENSC=SC+4
3052 IFV=6ORV=7THENPOKEK,8:IFZ=-1THENSC=SC+2
3055 IFV=1ANDZ=1THENH=0
3056 IFV=1ANDZ=-1THENF=0:SC=SC+10
3057 POKET3,0:POKET1+2,15:IFZ=-1THENGOSUB700
3058 IFV=0ANDZ=1THENM1=M1+1:GOSUB4000:POKE7679+M1,32:A=8142:IFM1=4THEN5000
3060 IFV=0ANDZ=-1THENM2=M2+1:GOSUB4100:POKE7698+M2,32:B=8163:IFM2=4THEN5100
3400 GOTO630
4000 POKEA,34:POKEA+22,42:POKEA+23,41
4010 FORS=7724TOA-66STEP22:POKES-22,32:POKES,33:POKES+22,31:FORT=1TO80:NEXTT,S
4020 POKEA-22,43:POKEA-21,44:POKEA+22,32:POKEA+23,32:FORT=1TO400:NEXT:POKEA-21,3
2
4030 FORS=A-66TO8186STEP22:POKES-22,32:POKES,33:POKES+22,31:FORT=1TO80:NEXTT,S:R
ETURN
4100 POKEB,34:POKEB+21,42:POKEB+22,41:SC=SC+20:GOSUB700
4110 FORS=7745TOB-66STEP22:POKES-22,32:POKES,33:POKES+22,31:FORT=1TO80:NEXTT,S
4120 POKEB-23,43:POKEB-22,44:POKEB+21,32:POKEB+22,32:FORT=1TO400:NEXT:POKEB-23,3
2
4130 FORS=B-66TO8207STEP22:POKES-22,32:POKES,33:POKES+22,31:FORT=1TO80:NEXTT,S:R
ETURN
5000 A$="------------------":PRINT"■■■■■■■  ■"A$"■■■■ ■G A M E  O V E R■ ■■■■"A$
5010 PRINT"■■■■■■■ AGAIN (Y/N) ? "
5020 WAIT197,64,64:SYS58648:IFPEEK(197)=11THENRUN
5040 END
5100 M2=0:PRINT"■■"SPC(19)"@@@":SC=SC+30:AM=60:ZX=ZX-.1:GOSUB700:GOTO450
5200 POKET3,0:AM=60:GOSUB700:V=0:Z=1:GOTO3058
```

Jumper



Der Archäologe Frank sucht nach wertvollen Gegenständen der antiken Stadt – Athen –.

Nachdem er einen verdeckten Eingang zu einer Höhle ausfindig machen kann, bereitet Frank die Fundstelle für die nötigen Ausgrabungen vor. Aufzüge werden montiert und die Höhlenwege erweitert. Nach den Vorbereitungen will Frank endlich die wertvollen Funde bergen. Aber – allerlei Gefahren und Hindernisse gefährden immer noch die Ausgrabungen. Gefährlicher Treibsand und große Termitenhaufen machen ihm am meisten zu schaffen. Doch, nachdem wertvolle Vasen ausfindig gemacht werden, will Frank sie auch bergen und mit dem Aufzug nach oben transportieren. Durch Berühren des Aufzuges fährt Frank nach oben. Nach unten kann er den Aufzug nicht benutzen, statt dessen springt er von Etage zu Etage. Das Spiel hat drei verschiedene Bilder. Wenn alle drei bewältigt wurden, beginnt das Spiel wieder mit dem ersten

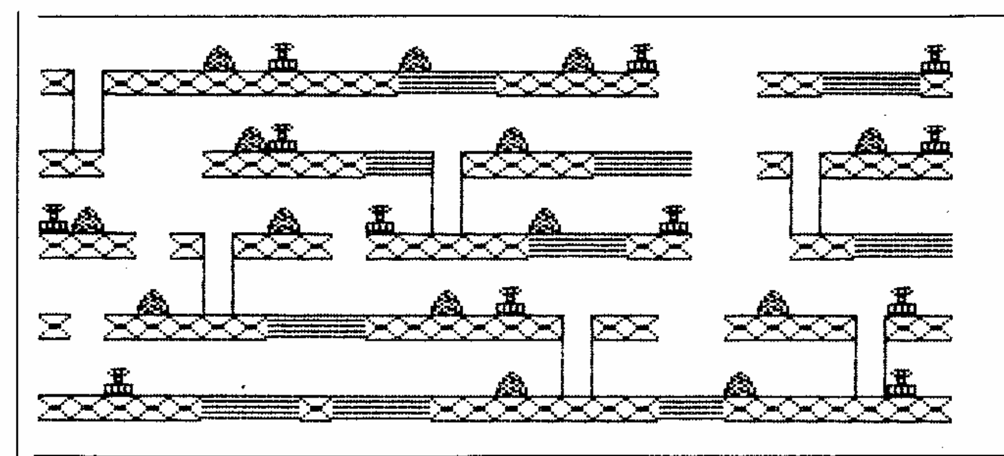


Bild. Die Steuerung ist wie folgt: ›J‹ *links,* ›K‹ *rechts,* ›I‹ *oben,* ›M‹ *unten.* ›Space‹ *ist springen.* Sie können Ihre Sprungrichtung durch: ›S‹ (links) und ›D‹ (rechts) verändern.
Die beiliegenden Daten müssen wie folgt eingegeben werden:

Teil 1

```
10 for I = 24576 to 24944
20 input A (eine Date)
25 poke I, A
30 next I
40 print CHR$ (4); „BSAVE JUMPER.
OBJ, A$ 6000, L 400"
```

Listing Teil 2

```
9/0/19/0/60/0/96/0/103/0/114/0/143/0/175/0/206/0/238/0/45/45/45/45/53/223/219/21
9/14/77/73/169/27/223/27/159/73/45/109/209/27/63/255/83/105/73/141/251/219/27/23
/77/73/73/62/63/63/63/63/6/0/13/13/13/13/254/219/219/110/73/73/254/219/219/110/1
3/13/13/254/31/31/31/110/73/73/254/219/219/110/73/73/254/31/31/31/6/0/50/14/45/3
2/28/55/0/241/14/30/14/30/14/30/14/30/6/0/9/45/141/27/31/191/9/45/141/223/223/10
/45/45/213/59/255/78/45/141/27/223/87/77/141/27/255/55/0/9/45/141/59/31/159/9/45
/141/223/223/10/45/45/213/59/255/78/45/141/251/251/74/77/21/255/255/2/0/73/137/0
/73/45/77/218/251/255/74/41/109/209/223/223/83/41/45/109/218/59/255/87/9/45/77/3
0/63/223/191/109/73/137/0/73/45/77/218/255/223/74/41/109/209/219/223/87/41/45/10
9/26/31/63/223/106/41/109/209/255/27/191/73/73/41/6/0/77/45/77/254/59/255/51/77/
45/77/62/223/223/55/13/45/45/13/254/59/255/51/77/45/77/254/255/255/110/77/105/62
/255/27/63/6/0/0/0/0/0/2/0/6/0/46/0/9/45/45/141/251/31/31/87/41/45/109/218/59/25
5/83/9/13/77/218/59/255/19/45/45/45/45/254/31/31/31/110/13/13/13/62/63/63/63/55/
0/73/73/73/218/27/255/155/73/109/77/218/31/255/159/9/109/45/141/59/31/255/191/10
5/77/13/21/31/31/255/63/46/13/13/45/13/62/255/59/255/6/0/0/1/
```

Listing Teil 3

```
0  REM  **************
1  REM  *---JUMPER---*
2  REM  *--COPYRIGHT-*
3  REM  *--1984 VON--*
4  REM  *-J O E R N--*
5  REM  *--R E D E R-*
6  REM  **************

9  GOSUB 12000
10  HOME : TEXT
11  POKE 768,173: POKE 769,48: POKE 770,192: POKE 771,136: POKE 772,208: POKE
    773,4: POKE 774,198: POKE 775,7
12  POKE 776,240: POKE 777,8: POKE 778,202: POKE 779,208: POKE 780,246: POKE
    781,166: POKE 782,6: POKE 783,76: POKE 784,0: POKE 785,3: POKE 786,9
```



```
   6
15  PRINT "BLOAD JUMPER.OBJ"
30  DIM A$(28,19): GOSUB 10000
50 A = 131:B = 1:A1 = 13:B1 = 0:M = 6
60  POKE 232,0: POKE 233,96: XDRAW 6 AT B,A
70 CR = 3
80 AS = 120
90 LV = 1
95 X =  FRE (0)
96 ET =  - 1
100 X =  PEEK ( - 16384)
110  IF X = 203 THEN  GOSUB 500
120  IF X = 202 THEN  GOSUB 600
125  IF X = 160 THEN  GOSUB 700
130  IF A$(B1,A1) = "5" THEN  GOSUB 1000
140  IF A$(B1,A1 + 1) = "3" THEN  GOSUB 600
150  IF A$(B1,A1 + 1) = "0" THEN  GOSUB 1100
160  IF SC = AS THEN  GOSUB 3000
170  IF X = 211 THEN RI =  - 1
180  IF X = 196 THEN RI = 1
185 FF =  FRE (0)
190  IF LV > 3 THEN  GOSUB 1600
200  GOTO 100
500  IF B1 = 27 THEN  RETURN
503 RI = 1
506  POKE  - 16368,0
510  IF A$(B1 + 1,A1) = "2" THEN 5000
520  IF A$(B1 + 1,A1) = "4" THEN  GOSUB 900:A$(B1 + 1,A1) = "0"
530  XDRAW M AT B,A
540 B = B + 10:B1 = B1 + 1:M = 6
550  XDRAW M AT B,A
560  RETURN
600  IF B1 = 0 THEN  RETURN
603 RI =  - 1
606  POKE  - 16368,0
610  IF A$(B1 - 1,A1) = "2" THEN 5000
620  IF A$(B1 - 1,A1) = "4" THEN  GOSUB 900:A$(B1 - 1,A1) = "0"
630  XDRAW M AT B,A
640 B = B - 10:B1 = B1 - 1:M = 5
650  XDRAW M AT B,A
660  RETURN
700  XDRAW M AT B,A
710 M = M + 2
720 B = B + RI * 10:A = A - 10:B1 = B1 + RI:A1 = A1 - 1
725  IF B1 > 27 THEN B1 = 27:B = 271
726  IF B1 < 0 THEN B1 = 0:B = 1
730  XDRAW M AT B,A
740  FOR I = 1 TO 2
750  XDRAW M AT B,A
760 B = B + RI * 10:B1 = B1 + RI
765  IF B1 > 27 THEN B1 = 27:B = 271
766  IF B1 < 0 THEN B1 = 0:B = 1
770  XDRAW M AT B,A
780  NEXT I
790  XDRAW M AT B,A
800 A = A + 10:B = B + RI * 10:A1 = A1 + 1:B1 = B1 + RI
805  IF B1 > 27 THEN B1 = 27:B = 271
806  IF B1 < 0 THEN B1 = 0:B = 1
```



```
810 M = M - 2
820  XDRAW M AT B,A
830  POKE  - 16368,0
833 S = 1
835  IF A$(B1,A1) = "4" THEN  GOSUB 900:A$(B1,A1) = "0"
836  IF A$(B1,A1) = "2" THEN 5000
837 S = 0
840  RETURN
900  POKE 232,24: POKE 233,97
905  IF S = 1 THEN  XDRAW 1 AT B - 1,A - 1:A$(B1,A1) = "0": GOTO 920
910  XDRAW 1 AT B + RI * 10 - 1,A - 1
920  POKE 6,50: POKE 7,5: CALL 768: POKE 6,200: POKE 7,5: CALL 768: POKE
     6,100: POKE 7,5: CALL 768: POKE 6,10: POKE 7,5: CALL 768: POKE 6,230
     : POKE 7,5: CALL 768
930 SC = SC + 10
940  POKE 232,0: POKE 233,96
950  RETURN
1000  XDRAW M AT B,A
1010  FOR I = A TO A - 20 STEP  - 10
1020  XDRAW 9 AT B,I
1030  FOR T = 1 TO 10: NEXT T
1035  POKE 6,A: POKE 7,5: CALL 768
1040  XDRAW 9 AT B,I
1045 A = I
1050  NEXT I
1070  XDRAW M AT B,I:A = I:A1 =  INT (A / 10)
1075 ET = ET + 1
1080  RETURN
1100  IF A = 131 THEN 5000
1103  XDRAW M AT B,A
1105  FOR I = A TO A + 20 STEP 10
1110  XDRAW M AT B,I
1115  POKE 6,180: POKE 7,3: CALL 768
1120  FOR T = 1 TO 10: NEXT T
1130  XDRAW M AT B,I
1135 A1 = A1 + 1
1140  NEXT I
1145 A = I
1146  XDRAW M AT B,A
1150  IF A$(B1,A1 + 1) = "0" THEN 5000
1155  IF A$(B1,A1) = "4" THEN S = 1: GOSUB 900:S = 0
1160  RETURN
1300  VTAB 22: HTAB 14 -  LEN ( STR$ (SC)): PRINT SC
1310  VTAB 22: HTAB 27 -  LEN ( STR$ (CR)): PRINT CR
1320  VTAB 22: HTAB 39 -  LEN ( STR$ (LV)): PRINT LV
1330  RETURN
1600  REM  BOMBE BEWEGEN
1610  IF X$ = "" THEN P = A - 1:L = 1:P1 = 0:Y1 = 0: IF ET / 2 =  INT (ET
      / 2) THEN P1 = 260:X$ = "X":Y1 =  - 10
1612  IF L = 1 THEN 1620
1615  GOTO 1630
1620 L = 0: IF P1 <  > 260 THEN P1 = 10:Y1 = 10:X$ = "X"
1626  XDRAW 3 AT P1,P
1630  REM
1640  XDRAW 3 AT P1,P
1650 X1 = A - 1: IF X1 <  > P THEN X$ = "": RETURN
1660 P1 = P1 + Y1
1665  IF P1 =  - 10 OR P1 = 280 THEN X$ = "": RETURN
```



```
1670  XDRAW 3 AT P1,P
1680  IF  INT (P1 / 10) = B1 AND  INT (P / 10) = A1 THEN 5000
1690  RETURN
3000  REM  LEVEL GESCHAFFT
3005 AS = AS + 320:SC = SC + 200: GOSUB 1300
3007 ET = 1
3010  FOR I = 255 TO 0 STEP  - 5
3020  POKE 6,I: POKE 7,5: CALL 768
3030  NEXT I: FOR I = 0 TO 255 STEP 5
3035  TEXT
3040  VTAB 22: HTAB 1: PRINT "  Score:00000    Credit:03    Level:00   "
3043 LV = LV + 1
3045  GOSUB 1300: FOR I = 1 TO 12: VTAB 24: PRINT : POKE 6,I * 10: POKE 7
     ,5: CALL 768: NEXT I: FOR I = 1 TO 3000: NEXT I
3046 LV = LV - 1
3050  GOSUB 10000
3060 A = 131:B = 1:A1 = 13:B1 = 0
3070  POKE 232,0: POKE 233,96
3080  XDRAW 6 AT B,A:M = 6
3090  RETURN
5000  IF A$(B1,A1 + 1) <  > "0" THEN 5050
5003 X$ = ""
5005  IF A = 131 THEN 5050
5010  XDRAW M AT B,A
5020 A = A + 10:A1 = A1 + 1
5025  POKE 6,A: POKE 7,5: CALL 768
5030  XDRAW M AT B,A
5040  GOTO 5000
5050  XDRAW M AT B,A
5060  ROT= 16:B = B + 10
5065  IF B = 281 THEN B = 271
5070  XDRAW 6 AT B,A
5080  FOR I = 1 TO 2000: NEXT I
5090 CR = CR - 1
5100  IF CR = 0 THEN 6000
5110  XDRAW 6 AT B,A: ROT= 0
5120 B = 1:A = 131:B1 = 0:A1 = 13
5130 M = 6: XDRAW M AT B,A
5140  POP : GOTO 100
6000  TEXT : HOME : VTAB 12: PRINT "      SCORE:";SC;"    LEVEL:";LV
6010  FOR I = 1 TO 5000: NEXT I
6020  RUN
10000  HGR2
10010  SCALE= 1: ROT= 0
10016 LV = LV + 1
10020  FOR T = 0 TO 14: FOR I = 0 TO 27
10030  READ A$(I,T)
10035  IF A$(I,T) = "0" THEN 10090
10040  IF A$(I,T) = "1" THEN  POKE 232,0: POKE 233,96: XDRAW 1 AT I * 10,
     T * 10 + 1: GOTO 10090
10050  IF A$(I,T) = "2" THEN  POKE 232,24: POKE 233,97: XDRAW 2 AT I * 10
     ,T * 10: GOTO 10090
10060  IF A$(I,T) = "3" THEN  FOR Q = T * 10 TO T * 10 + 9 STEP 2: HCOLOR=
     3: HPLOT I * 10,Q TO I * 10 + 9,Q: NEXT Q: GOTO 10090
10070  IF A$(I,T) = "4" THEN  POKE 232,24: POKE 233,97: XDRAW 1 AT I * 10
     ,T * 10: GOTO 10090
10080  HCOLOR= 3: HPLOT I * 10,T * 10 TO I * 10,T * 10 + 9: HPLOT I * 10 +
     9,T * 10 TO I * 10 + 9,T * 10 + 9
10090  NEXT I,T
```



```
10091  READ W: IF W =  - 99 THEN  RESTORE
10092  RETURN
10095  REM DATEN FUER BILD 1
10100  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10110  DATA  Ø,Ø,Ø,Ø,Ø,2,Ø,4,Ø,Ø,Ø,2,Ø,Ø,Ø,Ø,2,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,Ø,4
10120  DATA    1,5,1,1,1,1,1,1,1,1,1,3,3,3,1,1,1,1,1,Ø,Ø,Ø,1,1,3,3,3,1
10130  DATA  Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10140  DATA   Ø,5,Ø,Ø,Ø,Ø,2,4,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,4
10150  DATA  1,1,Ø,Ø,Ø,1,1,1,1,1,3,3,5,1,1,1,1,3,3,3,Ø,Ø,1,5,1,1,1,1
10160  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø
10170  DATA  4,2,Ø,Ø,Ø,Ø,Ø,2,Ø,Ø,4,Ø,5,Ø,Ø,2,Ø,Ø,Ø,4,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø
10180  DATA  1,1,1,Ø,1,5,1,1,1,Ø,1,1,1,1,1,3,3,3,1,1,Ø,Ø,Ø,1,1,3,3,3
10190  DATA  Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10200  DATA  Ø,Ø,Ø,2,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,Ø,Ø,4,Ø
10210  DATA  1,Ø,1,1,1,1,1,3,3,3,1,1,1,1,1,1,5,1,1,Ø,Ø,1,1,1,1,5,1,1
10220  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø
10230  DATA   Ø,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,5,Ø,Ø,Ø,Ø,2,Ø,Ø,Ø,5,4,Ø
10240  DATA  1,1,1,1,1,3,3,3,1,3,3,3,1,1,1,1,1,1,1,3,3,1,1,1,1,1,1,1
10245  DATA  5
10250  REM DATEN FUER BILD 2
10260  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10270  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,4,Ø,Ø,Ø,4,Ø,Ø,Ø,2,Ø,Ø,Ø,Ø
10280  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,1,1,1,1,1,1,3,3,3,1,1,1,1,1,1,1,1
10290  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10300  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø
10310  DATA  Ø,5,1,1,1,1,3,3,3,1,1,Ø,1,1,1,3,1,1,3,3,1,1,1,1,3,3,3,3
10320  DATA  Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10330  DATA  Ø,5,Ø,4,Ø,Ø,Ø,Ø,4,Ø,Ø,Ø,2,Ø,Ø,Ø,Ø,4,Ø,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø
10340  DATA   Ø,5,3,3,3,3,Ø,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,Ø,3,5,Ø
10350  DATA  Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø
10360  DATA   Ø,5,Ø,2,Ø,4,Ø,Ø,2,Ø,2,Ø,Ø,Ø,Ø,Ø,2,Ø,Ø,4,Ø,2,Ø,Ø,Ø,Ø,5,Ø
10370  DATA  1,1,1,3,3,3,1,1,1,1,3,3,1,1,5,1,1,1,1,1,1,3,3,3,1,1,1,Ø
10380  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10390  DATA  Ø,Ø,2,2,2,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,Ø,5,Ø,Ø,Ø,Ø,2,4,Ø,Ø,Ø,4,Ø,2,Ø
10400  DATA  1,1,1,1,1,1,3,3,3,1,1,1,1,1,3,3,1,1,1,1,1,1,1,1,1,1,1,1
10405  DATA  5
10410  REM DATEN FUER BILD 3
10420  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10430  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,4,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,4,Ø
10440  DATA  Ø,Ø,5,1,1,1,1,1,3,1,3,3,3,1,3,3,3,1,3,3,3,1,1,1,1,1,1,Ø
10450  DATA  Ø,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10460  DATA  Ø,Ø,5,Ø,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø
10470  DATA  Ø,Ø,3,3,3,1,1,1,1,Ø,Ø,Ø,1,1,1,Ø,3,3,3,3,3,1,1,1,1,5,Ø,Ø
10480  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø
10490  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø,4,Ø,Ø,2,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø
10500  DATA   Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,1,1,1,1,Ø,1,3,3,3,1,1,1,1,1,Ø,Ø,1,1,Ø,Ø
10510  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø
10520  DATA   Ø,Ø,4,Ø,Ø,2,Ø,5,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,Ø,Ø,4,Ø,Ø,2,Ø,Ø,Ø,Ø
10530  DATA  1,1,1,1,1,1,1,1,3,3,3,Ø,3,3,3,3,3,1,1,1,1,1,1,1,3,3,5,Ø
10540  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,5,Ø
10550  DATA  Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,Ø,4,Ø,Ø,Ø,Ø,Ø,Ø,2,Ø,Ø,Ø,Ø,Ø,2,Ø,5,Ø
10560  DATA   1,1,Ø,Ø,Ø,1,Ø,Ø,Ø,1,Ø,1,Ø,1,Ø,1,Ø,1,1,1,1,1,1,Ø,1,1,1,1
10565  DATA  -99                  #####
12000  HOME : TEXT                  #"
12010  VTAB 3: PRINT "              #"
12020  VTAB 4: PRINT "              # #   # # # #  ####   ###  # ##"
12030  VTAB 5: PRINT "              # #   # ## # # #    # #    # ##"
12040  VTAB 6: PRINT "              # #   # #  # # #    # ##### #"
12050  VTAB 7: PRINT "          #   # #   # #  # # #    # #       #"
12060  VTAB 8: PRINT "             ###   ###  #  # # ####    #### #"
12070  VTAB 9: PRINT "                                #
12075  VTAB 10: PRINT "                               #
```

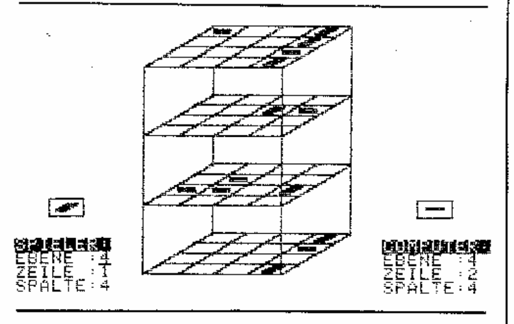


```
12080  VTAB 11: PRINT "                              #
12090  VTAB 12: PRINT "                              #
12100  VTAB 13: PRINT "                              #
12110  INVERSE : FOR I = 1 TO 40: VTAB 1: HTAB I: PRINT " ": VTAB 15: HTAB
     I: PRINT " ": NEXT I
12120  FOR I = 1 TO 15: VTAB I: HTAB 1: PRINT " ": VTAB I: HTAB 40: PRINT
     " ": NEXT I
12130  NORMAL : VTAB 17: PRINT "       Geschrieben von Joern Reder "
12140  VTAB 19: PRINT "          Copyright (C) 1 9 8 4"
12150  VTAB 21: PRINT "      Um zu spielen Taste druecken"
12160  VTAB 22: PRINT "      ----------------------------------"
12165  POKE  - 16368,0
12170  WAIT  - 16384,128
12180  RETURN
```

Tic – Tac – Toe

Tic – Tac – Toe, ein interessantes, dreidimensionales Strategiespiel für den Apple II, erfordert strategisches und logisches Denken.

Mit unserem ausgedruckten Programm ist es möglich, in dem interessanten dreidimensionalen Strategiespiel „Tic – Tac – Toe", teilweise auch bekannt als Quibic, gegen den Computer anzutreten. Daß der Apple II sich hervorragend dazu eignet, das dreidimensionale Spielfeld wirklichkeitsgetreu darzustellen, wird aus der abgebildeten Bildschirmgrafik ersichtlich.



Die Spielregeln sind recht einfach. Es geht darum, vier eigene Spielsteine in eine Reihe zu bringen. Dabei kann es sich um eine horizontale, waagerechte oder auch diagonale Linie handeln. Das heißt, auch räumliche Diagonalen sind zulässig. Man kann also eine Linie besetzen, die von der Ecke unten links (vorne) bis zu derjenigen oben rechts (hinten) reicht. Insgesamt existieren übrigens 76 solcher Linien, die einen Gewinn ermöglichen. Die Spielsteine werden abwechselnd auf dem Spielfeld plaziert, wobei ein Spielstein zu jeder Zeit auf einem beliebigen Feld plaziert werden kann. Es sei denn, dieses Feld ist schon besetzt. Das Spiel gewinnt seinen Reiz vor allem aus der Vielzahl der Möglichkeiten sowie aus der räumlichen Dimension, die dazu führt, daß man schon sehr gut aufpassen muß, wenn man nicht vom Computer überlistet werden will. Dieses Spiel schult eben auch auf recht interessante Weise das räumliche Vorstellungsvermögen.

Zur Bedienung des Programmes: Zu Beginn kann man die Spielstufe festlegen, auf der der Computer spielen soll. Dabei ist Stufe 9 diejenige mit der größten Spielstärke. Danach kann man entscheiden, wer das Spiel beginnen soll.

Bei Eingabe der Züge werden die Koordinaten eines jeden Feldes in Ebene, Zeile und Spalte zerlegt. Dabei gilt: Ebene 1 ist unten, Ebene 4 oben. Zeile 1 ist vorne, Zeile 4 hinten. Und Spalte 1 ist links, Spalte 4 rechts. Sollte man bei der Eingabe einer Koordinate einen Fehler gemacht haben, kann man mit dem Linkspfeil wieder zur betreffenden Eingabe zurückgelangen. Nach erfolgter Eingabe aller Koordinaten stellt der Computer dennoch eine Sicherheitsfrage, so daß man sich immer noch anders entscheiden kann.

```
1000  :
1010  :
1020  REM  DREIDIMENSIONALES TIC TAC TOE
1030  REM  (C)  ULF SCHLICHTMANN, 1984
1040  :
1050  :
2000  :
2010  :
2020  REM  INITIALISIEREN
2030  :
2040  :
2050  LOMEM: 16384
2060  DIM F(4,4,4): REM        SPIELFELD
2070  DIM W(76): REM           WERTE DER EINZELNEN
2080  DIM T(4,4,4,5): REM      TABELLE        ZUEGE
2090  DIM Z$(3): REM           KOORDINATEN
2100  DIM Z(1,3): REM          ZUEGE
2110 Z$(1) = "EBENE :"
2120 Z$(2) = "ZEILE :"
2130 Z$(3) = "SPALTE:"
2140  DIM V(13): REM           ZUGBEWERTUNG
2150  FOR I = 1 TO 13
```



```
2160  READ V(I)
2170  NEXT I
2180  DIM S(10): REM        LINIENBEWERTUNGEN EINES FELDES
2190  DIM M(10,3): REM      DIE ZEHN BESTEN ZUEGE
2200  DIM P(10): REM        DIE DAZUGEHOERIGEN ZUGBEWERTUNGEN
2210 :
2220  GOSUB 35000: REM      TABELLE EINLESEN
2230  GOSUB 36000: REM      SPIELFELD AUFBAUEN
2240 ZN =  - 1
5000 :
5010 :
5020  REM  EINFUEHRUNG
5030 :
5040 :
5050  HOME : VTAB 21: PRINT "WELCHE SCHWIERIGKEITSSTUFE (0-9)?": HTAB 20
5060  GET H$
5070  IF H$ < "0" OR H$ > "9" GOTO 5060
5080  PRINT H$;
5090 LE = 10 -  VAL (H$)
5100  HOME : VTAB 21: PRINT "WER SOLL BEGINNEN?"
5110  INVERSE : PRINT "C";
5120  NORMAL : PRINT "OMPUTER ODER ";
5130  INVERSE : PRINT "S";
5140  NORMAL : PRINT "PIELER ?    ";
5150  GET H$
5160  PRINT H$; CHR$ (8);
5170  IF H$ = "S" THEN  GOSUB 32000: GOSUB 33000: GOTO 10140
5180  IF H$ <  > "C" GOTO 5150
5190  GOSUB 32000: GOSUB 33000
10000 :
10010 :
10020  REM  HAUPTPROGRAMM
10030 :
10040 :
10050  GOSUB 15000: REM      GEWINNLINIEN CHECKEN
10060  GOSUB 17000: REM      SPIELERSIEG?
10070  IF S =  - 1 GOTO 26000
10080  GOSUB 25000: REM      UNENTSCHIEDEN?
10090  IF ZN = 64 GOTO 26000
10100  GOSUB 20000: REM      COMPUTERZUG ERMITTELN
10110  GOSUB 31020: REM      ZUG AUSGEBEN
10120  IF ZN = 64 GOTO 26000
10130  IF S = 3 GOTO 26000
10140  GOSUB 28000: REM      SPIELERZUG EINGEBEN
10150  GOSUB 30000: REM      ZUG AUSGEBEN
10160  IF ZN = 64 GOTO 26000
10170  GOTO 10050
15000 :
15010 :
15020  REM  WERTE DER GEWINNLINIEN FESTSTELLEN
15030 :
15040 :
15050  FOR A = 1 TO 4: FOR B = 1 TO 4
15060 W((A - 1) * 10 + B) = F(A,B,1) + F(A,B,2) + F(A,B,3) + F(A,B,4)
15070  NEXT B
15080  FOR C = 1 TO 4
15090 W((A - 1) * 10 + 4 + C) = F(A,1,C) + F(A,2,C) + F(A,3,C) + F(A,4,C)
15100  NEXT C
```



```
15110 W((A - 1) * 10 + 9) = F(A,1,1) + F(A,2,2) + F(A,3,3) + F(A,4,4)
15120 W(A * 10) = F(A,1,4) + F(A,2,3) + F(A,3,2) + F(A,4,1)
15130  NEXT A
15140  FOR B = 1 TO 4: FOR C = 1 TO 4
15150 W(40 + (B - 1) * 4 + C) = F(1,B,C) + F(2,B,C) + F(3,B,C) + F(4,B,C)
15160  NEXT C: NEXT B
15170 W(57) = F(1,4,1) + F(2,3,2) + F(3,2,3) + F(4,1,4)
15180 W(58) = F(1,4,4) + F(2,3,3) + F(3,2,2) + F(4,1,1)
15190 W(59) = F(1,1,1) + F(2,2,2) + F(3,3,3) + F(4,4,4)
15200 W(60) = F(1,1,4) + F(2,2,3) + F(3,3,2) + F(4,4,1)
15210  FOR B = 1 TO 4
15220 W(60 + B) = F(1,B,1) + F(2,B,2) + F(3,B,3) + F(4,B,4)
15230 W(64 + B) = F(1,B,4) + F(2,B,3) + F(3,B,2) + F(4,B,1)
15240  NEXT B
15250  FOR C = 1 TO 4
15260 W(68 + C) = F(1,1,C) + F(2,2,C) + F(3,3,C) + F(4,4,C)
15270 W(72 + C) = F(1,4,C) + F(2,3,C) + F(3,2,C) + F(4,1,C)
15280  NEXT C
15290  RETURN
17000 :
17010 :
17020  REM  SPIELERSIEG?
17030 :
17040 :
17050  FOR I = 1 TO 76
17060  IF W(I) =  - 4 THEN S =  - 1
17070  NEXT I
17080  RETURN
20000 :
20010 :
20020  REM  COMPUTERZUG BERECHNEN
20030 :
20040 :
20050  FOR I = 1 TO 10:P(I) =  - 99:M(I,1) = 0: NEXT I
20060  FOR A = 1 TO 4: FOR B = 1 TO 4: FOR C = 1 TO 4
20070  IF F(A,B,C) <  > 0 GOTO 20340
20080 S(1) = W(40 + (B - 1) * 4 + C)
20090 S(2) = W((A - 1) * 10 + B)
20100 S(3) = W((A - 1) * 10 + 4 + C)
20110  FOR I = 1 TO T(A,B,C,5)
20120 S(I + 3) = W(T(A,B,C,I))
20130  NEXT I
20140 L = 0:J = 0
20150 P =  INT (T(A,B,C,5) / 2)
20160  FOR I = 1 TO T(A,B,C,5) + 3
20170 P = P + V(S(I) + 4)
20180  IF S(I) = 4 THEN L = L + 1: IF L = 2 THEN P = P + 20
20190  IF S(I) = 5 THEN J = J + 1: IF J = 2 THEN P = P + 40
20200  NEXT I
20210 :
20220  FOR I = 1 TO 10
20230  IF P > P(I) GOTO 20270
20235  IF P = P(I) THEN  IF  RND (P(I)) > .5 GOTO 20270
20240  NEXT I
20250  GOTO 20340
20260 :
20270  IF I = 10 GOTO 20320
20280  FOR U = I TO 9
```



```
20290 P(U + 1) = P(U)
20300 M(U + 1,1) = M(U,1):M(U + 1,2) = M(U,2):M(U + 1,3) = M(U,3)
20310  NEXT U
20320 P(I) = P:M(I,1) = A:M(I,2) = B:M(I,3) = C
20330 :
20340  NEXT C: NEXT B: NEXT A
20350 :
20360 SU = P( INT (LE / 2) + 1)
20395 SU =  INT (SU / LE)
20400 SI =  INT ( RND (P(1)) * SU) + 1
20405 Z =  - 1
20410  FOR U = 2 TO 10
20420  IF P(U) < SI THEN Z = U - 1:U = 10
20430  NEXT U
20440  IF Z =  - 1 THEN Z =  INT ( RND (P) * LE) + 1
20445  IF Z = 0 THEN Z = 1
20447  IF LE = 1 THEN Z = 1
20450  IF M(Z,1) = 0 GOTO 20400
20460 Z(0,1) = M(Z,1):Z(0,2) = M(Z,2):Z(0,3) = M(Z,3)
20470  IF P(Z) > 4900 THEN S = 3
20480  RETURN
25000 :
25010 :
25020  REM  UNENTSCHIEDEN?
25030 :
25040 :
25050  FOR I = 1 TO 76
25060  IF W(I) <  > 8 AND W(I) <  > 4 GOTO 25090
25070  NEXT I
25080 ZN = 64
25090  RETURN
26000 :
26010 :
26020  REM  SPIELENDE
26030 :
26040 :
26050  HOME
26060  VTAB 21
26070  INVERSE
26080  IF S =  - 1 THEN  PRINT "GRATULIERE, SIE HABEN GEWONNEN."
26090  IF S = 3 THEN  PRINT "HURRA, ICH HABE GEWONNEN."
26100  IF S = 0 THEN  PRINT "DAS SPIEL ENDETE DIESMAL UNENTSCHIEDEN."
26110  NORMAL
26120  END
28000 :
28010 :
28020  REM  SPIELERZUG EINGEBEN LASSEN
28030 :
28040 :
28050  FOR I = 1 TO 3
28060  VTAB 21 + I: HTAB 8
28070  GET H$
28080  IF H$ =  CHR$ (8) THEN  IF I > 1 THEN I = I - 1: GOTO 28060
28090  IF H$ < "1" OR H$ > "4" GOTO 28070
28100  PRINT H$;
28110 Z(1,I) =  VAL (H$)
28120  NEXT I
28122  IF F(Z(1,1),Z(1,2),Z(1,3)) = 0 GOTO 28130
```



```
28124  PRINT  CHR$ (7) CHR$ (7) CHR$ (7);
28126  GOTO 28050
28130  VTAB 24: HTAB 10: INVERSE : PRINT "OK (J/N)";: NORMAL
28140 I1 = 3
28142  HCOLOR= I1: GOSUB 30070
28144 I1 = 3 - I1
28146 H =  PEEK (49152)
28148  IF H < 128 GOTO 28142
28150  IF H <  > 202 AND H <  > 206 THEN H =  PEEK (49168): GOTO 28142
28152  IF I1 = 0 THEN  HCOLOR= 0: GOSUB 30070
28160  IF H = 206 GOTO 28050
28170  HTAB 10: PRINT "          ";
28200  HCOLOR= 3
28210  RETURN
30000 :
30010 :
30020  REM  SPIELFELD UPDATEN
30030 :
30040 :
30050  GOSUB 32000: REM  SPIELERZUG
30060 F(Z(1,1),Z(1,2),Z(1,3)) =  - 1
30070 PH = 75 + Z(1,3) * 20 + (Z(1,2) - 1) * 10
30080 PV = 33 + (4 - Z(1,1)) * 40 - Z(1,2) * 6
30090  FOR U = PH + 1 TO PH + 7
30100  HPLOT U,PV - 2 TO U - 7,PV + 2
30110  NEXT U
30120  RETURN
31000 :
31010 :
31020  GOSUB 32000: REM  COMPUTERZUG
31030 F(Z(0,1),Z(0,2),Z(0,3)) = 3
31040 PH = 75 + Z(0,3) * 20 + (Z(0,2) - 1) * 10
31050 PV = 33 + (4 - Z(0,1)) * 40 - Z(0,2) * 6
31060  FOR U = PV - 1 TO PV + 1
31070  HPLOT PH - 5,U TO PH + 5,U
31080  NEXT U
31090  RETURN
32000 :
32010 :
32020  REM  ZUEGE AUSGEBEN
32030 :
32040 :
32050  HOME : VTAB 21
32060  INVERSE
32070  PRINT "SPIELER:";: NORMAL : PRINT "                                        ";: INVERSE
     : PRINT "COMPUTER:";: NORMAL
32080  FOR I = 1 TO 3
32090  PRINT : PRINT Z$(I);Z(1,I); TAB( 31);Z$(I);Z(0,I);
32100  NEXT I
32110 ZN = ZN + 1
32120  RETURN
33000 :
33010 :
33020  REM  FELDMARKER DEMONSTRIEREN
33030 :
33040 :
33050  FOR U = 31 TO 37
33060  HPLOT U,140 TO U - 7,144
33070  NEXT U
```

```
33080  FOR U = 141 TO 142
33090  HPLOT 235,U TO 245,U
33100  NEXT U
33110  FOR U = 20 TO 230 STEP 210
33120  HPLOT U,137 TO U + 20,137 TO U + 20,147 TO U,147 TO U,137
33130  NEXT U
33140  RETURN
35000 :
35010 :
35020  REM  TABELLE EINLESEN
35030 :
35040 :
35050  FOR A = 1 TO 4: FOR B = 1 TO 4: FOR C = 1 TO 4
35060 T(A,B,C,5) = 4
35070  FOR I = 1 TO 5
35080  IF I = 5 GOTO 35130
35090  READ H$
35100  IF H$ = "#" THEN I = 5:H = 1
35110  IF H$ <  > "#" THEN H =  VAL (H$)
35120 T(A,B,C,I) = H
35130  NEXT I: NEXT C: NEXT B: NEXT A
35140  RETURN
36000 :
36010 :
36020  REM  SPIELFELD ZEICHNEN
36030 :
36040 :
36050  TEXT : HOME : HGR
36060  HCOLOR= 3:LM = 80
36070  FOR I = 30 TO 150 STEP 40
36080  FOR U = LM TO LM + 80 STEP 20
36090  HPLOT U,I TO U + 40,I - 24
36100  NEXT U
36110  FOR U = 0 TO 24 STEP 6
36120  HPLOT LM + U * 8 / 5 + 1,I - U TO LM + 80 + U * 8 / 5 + 1,I - U
36130  NEXT U: NEXT I
36140  HCOLOR= 2
36150  HPLOT LM,30 TO LM,150: HPLOT LM + 80,30 TO LM + 80,150: HPLOT LM +
     120,6 TO LM + 120,125
36160  FOR U = 30 TO 110 STEP 40: HPLOT LM + 40,U TO LM + 40,U + 15: NEXT
     U
36170  HCOLOR= 3
36180  RETURN
50000 :
50010  REM  DATEN FUER P(I) (ZUGBEWERTUNG)
50020 :
50030  DATA 81,4,-1,0,-1,-1,2,0,-1,5,0,0,5000
50040 :
50050  REM  DATEN FUER T(A,B,C,I)
50060 :
50070  DATA 59,69,61,9,70,#,71,#,65,10,60,72,62,#,9,#,10,#,66,#
50080  DATA 63,#,10,#,9,#,67,#,64,10,51,73,74,#,75,#,68,76,9,58
50090  DATA 19,#,61,#,65,#,20,#,69,#,70,59,19,62,71,60,66,20,72,#
50100  DATA 73,#,74,20,57,63,75,19,58,67,76,#,20,#,64,#,68,#,19,#
50110  DATA 29,#,65,#,61,#,30,#,73,#,74,29,66,58,75,30,62,57,76,#
50120  DATA 69,#,70,30,60,67,71,29,63,59,72,#,30,#,68,#,64,#,29,#
50130  DATA 39,65,73,58,74,#,75,#,40,61,76,57,66,#,39,#,40,#,62,#
50140  DATA 67,#,40,#,39,#,63,#,68,69,40,60,70,#,71,#,64,72,39,59
```

Eine Basic-Erweiterung für den Commodore 64 mit 14 Befehlen:

Scroll – verschiebt den Bildschirminhalt nach links ›L‹, rechts ›R‹, oben ›U‹ und unten ›D‹.
CLS – löscht den Bildschirm.
Home – setzt den Cursor in die linke obere Ecke.
Inverse – schaltet auf Inversen Zeichensatz um.
Normal – schaltet wieder auf normalen Zeichensatz um.
HTAB ... – positioniert den Cursor horizontal.
VTAB ... – positioniert den Cursor vertikal.
OLD – holt mit ›NEW‹ gelöschtes Programm zurück.

MERGE – lädt ein Programm hinter ein bereits vorhandenes Programm. Die Zeilennummern müssen dann jedoch größer sein.
SPRITE – setzt den SPRITE mit der Nummer ›n‹ und der Farbe ›c‹ auf die Position x/y.
Danach kann noch nach belieben angefügt werden:

›P‹ für Priorität
›←‹ für x-Streckung
›↑‹ für y-Streckung
›↑←‹ oder ›←↑‹ für x- und y-Streckung

Beispiel:

SPRITE 3, (200, 180), 1, P, ←↑
Spritenummer 3 wird weiß (1) auf die Position 200, 180 mit Priorität und x/y-Streckung gesetzt.

Beispiel:

SPRITE 0. (270, 50), 0, ←, P
Spritenummer ›0‹ wird schwarz auf die Position 270/50 mit Priorität und x-Streckung gesetzt.

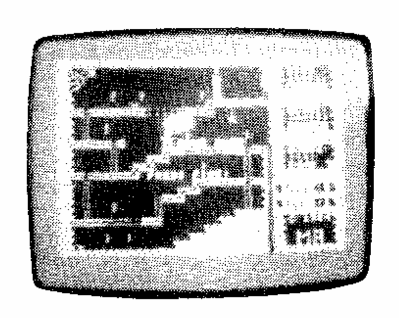
CLSPR – löscht den SPRITE mit der Nummer ›n‹

```
0 DATA85,32,115,0,201,76,240,77,234,234,234,234,234,234,234,234,234,234
1 DATA234,234,234,234,234,234,234,234,201,82,208,18,76,191,160,234,234
2 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,201,85,208,3
3 DATA76,5,161,201,68,208,3,76,26,161,162,11,76,19,194,234,234,234,234
4 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,169,192,133
5 DATA250,169,7,133,251,169,192,133,252,169,219,133,253,234,234,234,234
6 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,160,1,177
7 DATA250,136,145,250,200,177,252,136,145,252,200,200,192,40,208,239,165
8 DATA250,56,233,40,133,250,165,251,233,0,133,251,165,252,56,233,40,133
9 DATA252,165,253,233,0,133,253,165,251,201,4,16,205,76,170,193,234,234
10 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,162
11 DATA192,160,7,134,250,132,251,162,192,160,219,134,252,132,253,160,38
12 DATA177,250,200,145,250,136,177,252,200,145,252,136,136,192,255,208,239
13 DATA165,250,56,233,40,133,250,165,251,233,0,133,251,165,252,56,233,40
14 DATA133,252,165,253,233,0,133,253,165,251,201,3,208,205,76,170,193,76
15 DATA53,194,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
16 DATA234,234,234,162,191,160,7,134,250,132,251,162,191,160,219,134,252
17 DATA132,253,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
18 DATA234,234,234,160,0,177,250,160,40,145,250,160,0,177,252,160,40,145
19 DATA252,165,250,56,233,1,133,250,165,251,233,0,133,251,165,252,56,233
20 DATA1,133,252,234,234,234,234,234,234,234,234,234,234,234,234,234,234
21 DATA234,234,234,234,165,253,233,0,133,253,165,251,201,3,208,190,76,170
22 DATA193,32,115,0,32,70,194,32,121,0,201,137,240,5,169,167,32,87,194,165
23 DATA97,208,9,32,121,194,32,138,194,76,173,193,32,121,0,176,6,32,104,194
24 DATA76,173,193,165,122,208,20,234,234,234,234,234,234,234,234,234,234
25 DATA234,234,234,234,234,234,234,234,198,123,198,122,76,155,194,165,43
26 DATA164,44,133,254,132,255,160,3,200,177,254,208,251,234,234,234,234
27 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,200,152,24
28 DATA101,254,160,0,145,43,165,255,105,0,200,145,43,136,162,3,230,254,208
29 DATA2,230,255,177,254,208,244,202,208,243,165,254,105,2,133,45,165,255
30 DATA105,0,133,46,165,55,164,56,133,51,132,52,165,45,234,234,234,234,234
31 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,164,46,133,45
32 DATA132,46,133,47,132,48,133,49,132,50,76,170,193,32,115,0,32,36,194
33 DATA142,239,207,138,24,105,248,144,5,162,14,76,19,194,169,1,224,0,240
34 DATA5,10,202,76,88,162,141,6,207,173,21,208,13,6,207,141,21,208,169,44
35 DATA32,87,194,169,40,32,87,194,32,186,194,32,203,194,165,21,201,1,240
36 DATA9,201,0,240,17,162,14,76,19,194,173,16,208,13,6,207,141,16,208,76
```

```
37 DATA165,162,173,16,208,73,255,13,6,207,73,255,141,16,208,173,239,207
38 DATA10,168,165,20,153,0,208,169,44,32,87,194,32,36,194,173,239,207,10
39 DATA168,138,153,1,208,169,41,32,87,194,169,44,32,87,194,32,36,194,138
40 DATA172,239,207,153,39,208,32,121,0,201,44,240,12,32,210,163,32,182,163
41 DATA32,196,163,76,173,193,32,115,0,201,80,240,13,201,95,240,68,201,174
42 DATA240,106,162,11,76,19,194,32,168,163,32,115,0,201,44,240,3,76,147
43 DATA163,32,115,0,201,95,208,19,32,220,163,32,115,0,201,174,240,3,76,156
44 DATA163,32,230,163,76,170,193,201,174,208,208,32,230,163,32,115,0,201
45 DATA95,208,114,32,220,163,76,170,193,32,220,163,32,115,0,201,174,208
46 DATA23,32,230,163,32,115,0,201,44,208,67,32,115,0,201,80,208,165,32,168
47 DATA163,76,170,193,32,196,163,32,121,0,76,70,163,32,230,163,32,115,0
48 DATA201,95,208,26,32,220,163,32,115,0,201,44,208,25,32,115,0,201,80,240
49 DATA3,76,246,162,32,168,163,76,170,193,32,182,163,32,121,0,76,112,163
50 DATA32,210,163,76,173,193,32,182,163,32,196,163,76,173,193,32,196,163
51 DATA76,173,193,32,182,163,76,173,193,173,27,208,73,255,13,6,207,73,255
52 DATA141,27,208,96,173,29,208,73,255,13,6,207,73,255,141,29,208,96,173
53 DATA23,208,73,255,13,6,207,73,255,141,23,208,96,173,27,208,13,6,207,141
54 DATA27,208,96,173,29,208,13,6,207,141,29,208,96,173,23,208,13,6,207,141
55 DATA23,208,96,32,115,0,32,36,194,138,24,105,248,144,23,162,14,76,19,194
56 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
57 DATA234,169,1,224,0,240,5,10,202,76,21,164,141,6,207,173,21,208,73,255
58 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
59 DATA234,13,6,207,73,255,141,21,208,76,173,193,169,147,32,220,194,76,170
60 DATA193,169,19,32,220,194,76,170,193,234,234,234,234,234,234,234,234
61 DATA234,234,234,234,234,234,234,234,234,234,169,18,32,220,194,76,170
62 DATA193,169,146,32,220,194,76,170,193,32,115,0,32,36,194,138,24,105,216
63 DATA144,5,162,14,76,19,194,138,168,166,214,32,237,194,76,173,193,32,115
64 DATA0,32,36,194,138,24,105,231,144,5,162,14,76,19,194,138,168,164,211
65 DATA32,237,194,76,173,193,165,45,56,233,2,170,165,46,233,0,168,169,1
66 DATA133,186,169,0,32,46,195,138,24,105,2,133,45,152,105,0,133,46,32,29
67 DATA195,76,170,193,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
68 DATA0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,125,203,220,55,1
69 DATA128,5,55,25,5,128,139,44,43,178,255,9,157,230,213,135,255
70 DATA102,98,204,217,55,27,49,15,35,85,75,18,19,80,205,174,98
71 DATA220,78,169,0,208,1,80,179,62,0,255,5,69,72,148,194,0
72 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,201,85,208,3
73 DATA76,5,161,201,68,208,3,76,26,161,162,11,76,19,194,234,234,234,234
74 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,169,192,133
75 DATA250,169,7,133,251,169,192,133,252,169,219,133,253,234,234,234,234
76 DATA32,0,192,88,76,134,227,234,234,234,234,234,234,234,234,234,234,234
77 DATA234,234,234,234,234,234,234,234,165,122,141,4,207,165,123,141,5,207
78 DATA220,78,169,0,208,1,80,179,62,0,255,5,69,72,148,194,0
79 DATA-1
91 DATA169,188,141,8,3,169,192,141,9,3,234,234,234,234,234,234,234,234,234
92 DATA234,234,234,234,234,234,234,234,234,234,169,195,162,194,160,205,141
93 DATA4,128,142,5,128,140,6,128,162,56,160,48,142,7,128,140,8,128,234,234
94 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
95 DATA162,113,160,192,142,0,128,140,1,128,234,234,234,234,234,234,234,234
96 DATA234,234,234,234,234,234,234,234,234,234,234,162,94,160,254,142,2
97 DATA128,140,3,128,96,142,22,208,32,163,253,32,80,253,234,234,234,234
98 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,32,21
99 DATA253,32,91,255,32,83,228,32,191,227,32,55,195,32,68,166,162,251,154
100 DATA32,0,192,88,76,134,227,234,234,234,234,234,234,234,234,234,234,234
101 DATA234,234,234,234,234,234,234,234,165,122,141,4,207,165,123,141,5,207
102 DATA162,0,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
103 DATA234,234,234,234,173,4,207,133,122,173,5,207,133,123,169,1,141,255
104 DATA207,189,179,193,240,36,32,115,0,221,179,193,240,24,234,234,234,234
105 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,234,234,169,0
```

```
106 DATA141,255,207,232,76,234,192,232,189,179,193,208,38,232,189,179,193
107 DATA208,33,234,234,234,234,234,234,234,234,234,234,234,234,234,234,234
108 DATA234,234,234,234,173,4,207,133,122,173,5,207,133,123,76,228,167,232
109 DATA232,173,255,207,240,149,234,234,234,234,234,234,234,234,234,234,234
110 DATA234,234,234,234,234,234,234,234,202,202,189,179,193,141,0,207,232
111 DATA189,179,193,141,1,207,32,129,193,108,0,207,234,234,234,234,234,234
112 DATA234,234,234,234,234,234,234,234,234,234,234,234,234,8,72,165,1,41
113 DATA254,133,1,104,40,96,234,234,234,234,234,234,234,234,234,234,234,234
114 DATA234,234,234,234,234,234,234,8,72,165,1,9,1,133,1,104,40,96,32,115
115 DATA0,32,159,193,76,174,167,83,67,82,79,76,76,0,1,160,139,0,129,161,143
116 DATA0,151,161,79,76,68,0,200,161,83,80,82,73,84,69,0,66,162,67,76,83
117 DATA80,82,0,240,163,67,76,83,0,67,164,72,79,77,69,0,75,164,73,78,86,69
118 DATA82,83,69,0,101,164,78,176,77,65,76,0,109,164,72,84,65,66,0,117,164
119 DATA86,84,65,66,0,144,164,77,69,82,71,69,0,171,164,0,0,0,32,159,193,32
120 DATA58,164,76,134,227,234,234,234,234,234,234,234,234,32,159,193,32,158
121 DATA183,76,129,193,234,234,234,234,234,234,234,234,32,159,193,32,234
122 DATA232,76,170,193,234,234,234,234,234,234,234,234,32,159,193,32,158
123 DATA173,76,129,193,234,234,234,234,234,234,234,234,32,159,193,32,255
124 DATA174,76,129,193,234,234,234,234,234,234,234,234,32,159,193,32,160
125 DATA168,76,129,193,234,234,234,234,234,234,234,234,32,159,193,32,9,169
126 DATA76,129,193,234,234,234,234,234,234,234,234,32,159,193,32,251,168
127 DATA76,129,193,234,234,234,234,234,234,234,234,32,159,193,76,188,192
128 DATA234,234,234,234,234,234,234,234,32,159,193,32,96,166,76,129,193,234
129 DATA234,234,234,234,234,234,234,32,159,193,32,138,173,76,129,193,234
130 DATA234,234,234,234,234,234,234,32,159,193,32,247,183,76,129,193,234
131 DATA234,234,234,234,234,234,234,32,159,193,32,210,255,76,129,193,234
132 DATA234,234,234,234,234,234,234,32,159,193,32,10,229,76,129,193,234,234
133 DATA234,234,234,234,234,234,32,159,193,32,19,166,76,129,193,234,234,234
134 DATA234,234,234,234,234,32,159,193,76,169,164,234,234,234,234,234,234
135 DATA234,234,32,159,193,32,51,165,76,129,193,234,234,234,234,234,234,234
136 DATA234,32,159,193,32,158,244,76,129,193,169,63,160,195,32,30,171,96
137 DATA147,13,32,32,32,32,42,42,42,42,32,32,32,32,32,84,72,69,32,32,32,32
138 DATA66,65,83,73,67,32,65,83,32,32,32,42,42,42,42,32,32,32,32,13,13,32
139 DATA32,32,32,32,32,40,67,41,32,49,57,56,52,32,66,89,32,83,84,69,80,72
140 DATA65,78,32,66,65,82,84,79,83,67,72,13,13,32,54,52,75,32,82,65,77,32
141 DATA83,89,83,84,69,77,32,51,56,57,49,49,32,66,65,83,73,67,32,66,89,84
142 DATA69,83,32,70,82,69,69,13,13,0,255,232,40,16,243,96,162,65,32,162,195
143 DATA162,41,32,175,195,32,146,193,76,127,192,162,45,32,162,195,48,246
144 DATA32,14,193,32,29,193,144,9,133,93,32,36,193,133,92,176,94,169,63,32
145 DATA210,255,208,222,169,32,164,211,136,145,209,32,109,193,144,237,32
146 DATA212,195,32,14,193,201,45,240
150 DATA-2
170 :REM ******************************
180 :REM ***                        ***
190 :REM ***    EXTENDED BASIC AS   ***
200 :REM ***       (C) 1984 BY      ***
210 :REM ***    STEPHAN BARTOSCH    ***
235 :REM ***                        ***
240 :REM ******************************
690 :
700 :REM***** EINLESEN DES ERSTEN *****
710 :REM*****       TEILS         *****
740 I=10*16↑3
760 READ A
780 IF A=-1 THEN 1000
800 POKE I , A
820 I=I+1
```

Top



```
The Basic – letzter Teil
840 GOTO 760
1000 REM**** EINLESEN DES ZWEITEN ****
1010 REM****          TEILS          ****
1040 I=12*16↑3
1060 READ A
1080 IF A=-2 THEN 2000
1100 POKE I , A
1120 I=I+1
1140 GOTO 1060
2000 :REM*** START DER ERWEITERUNG   ***
2010 :REM*** DURCH SYS 49152 UND     ***
2020 :REM*** ANSCHLIESSENDES RESET   ***
2030 :REM*** (D.H. SYS 64738 !)      ***
2070 SYS 49152:REM   INITIALISIERUNG
2090 SYS 64738:REM   RESET
```

programm

Ein Topprogramm, ausgewählt von der Redaktion zum besten Programm in unserer Ausgabe. Eine Supergraphik zeichnet dieses Spiel im besonderen aus.

Ziel des Spiels ist es, zwei Spinnen, die auf ihren Netzen herumkrabbeln, zu beseitigen.

Vernichten kann man beide Spinnen allerdings nur mit einem Insektengift. Eine Spraydose dieses Giftes liegt noch im Keller, der zu allem Übel aber verschlossen ist. Also, zunächst einmal müssen alle Schlüssel eingesammelt werden. Zehn Stück sind es im Ganzen. Sind alle Schlüssel gefunden, muß der rote Knopf auf der obersten Leiter gedrückt werden. Erschwert wird der Weg zur Leiter durch kleine herumkrabbelnde Spinnen, die aber mit viel Geschick und guter Reaktion übersprungen werden können. Wird die Figur von einer der kleinen Spinnen berührt, so fällt sie vom Gerüst und das Spiel beginnt mit einer neuen Figur. Sind Sie zur Spraydose gelangt, muß die Dose berührt werden und das Spray entfaltet seine Wirkung. Für Sie bedeutet dies: der nächste Level.

Gespielt wird mit dem Joystick an Port 2. Und wir wünschen viel Vergnügen.

```
1 POKE56,56:GOTO10
2 ********************
3 *  S P I D E R S  *
4 *  COPYRIGHT '84  *
5 *  BY  MR - SOFT  *
6 *  RONALD  MAYER  *
7 *                 *
8 *  A U S T R I A  *
9 ********************
10 DATA169,0,133,252,169,56,133,253
20 DATA169,0,133,254,169,208,133,255
30 DATA162,0,160,0,177,254,145,252
40 DATA200,208,249,232,224,8,240,7
50 DATA230,253,230,255,76,20,128,96
60 FORI=32768TO32807:READA:POKEI,A:NEXT
70 POKE56334,0:POKE1,51:SYS32768
80 POKE1,55:POKE56334,1:POKE53272,30
90 V=53248:RI=2040
100 POKEV+33,8:POKEV+32,8
110 PRINT"▒JUST A MOMENT..."
120 POKEV+27,2:POKEV+28,1:POKEV+37,6
130 POKEV+38,8:POKEV+39,5:POKEV+40,2
140 POKERI+1,251:GOTO25000
160 J=PEEK(56320):SYS37225:BO=BO-1
161 IFPEEK(X-40)=60THENGOSUB600
162 IFPEEK(X+40)<38THEN300
```

```
163 X1=PEEK(V):X2=PEEK(V+1)
164 IFPEEK(X)<32ORPEEK(X-40)<32THEN300
165 IFRND(1)>1-RU/100THENPOKE1959,28
170 IF(JAND16)=0THEN240
172 IF(JAND1)=0THEN220
174 IF(JAND2)=0THEN230
176 IF(JAND4)=0THEN210
178 IF(JAND8)=0THEN200
179 IFX=1969THEN400
180 FORI=1TO80:NEXT:GOTO160
200 POKERI,248:IFPEEK(X+1)>39THEN180
202 POKEV,X1+4:FORI=1TO60:NEXT
204 POKERI,247:POKEV,X1+8:X=X+1:GOTO160
210 POKERI,246:IFPEEK(X-1)>39THEN180
212 POKEV,X1-4:FORI=1TO60:NEXT
214 POKERI,245:POKEV,X1-8:X=X-1:GOTO160
220 IFPEEK(X)<>39THEN180
221 POKERI,249:X2=X2-4:POKEV+1,X2
222 FORI=1TO50:NEXT:POKERI,250:X2=X2-4:POKEV+1,X2:X=X-40:GOTO160
230 IFPEEK(X+40)<>39THEN180
231 POKERI,249:X2=X2+4:POKEV+1,X2
232 FORI=1TO50:NEXT:POKERI,250:X2=X2+4:POKEV+1,X2:X=X+40:GOTO160
240 IFPEEK(RI)>248THEN180
241 IFPEEK(RI)>246THEN251
242 IFPEEK(X-42)>159THEN180
243 X2=X2-8:X1=X1-2:X=X-40:POKEV,X1:POKEV+1,X2:POKERI,245
244 X2=X2-4:X1=X1-6:X=X-1:POKEV,X1:POKEV+1,X2:IFPEEK(X+40)<30THENGOSUB280
245 IFPEEK(X)=38THENGOSUB290
246 X2=X2+4:X1=X1-6:X=X-1:POKEV,X1:POKEV+1,X2:POKERI,246
247 IFPEEK(X+40)>159THENX1=X1-2:POKEV,X1:GOTO160
248 X2=X2+8:X1=X1-2:X=X+40:POKEV,X1:POKEV+1,X2:IFPEEK(X+40)>159THEN160
249 X2=X2+8:POKEV+1,X2:X=X+40:IFPEEK(X+40)>159THEN160
250 GOTO300
251 IFPEEK(X-38)>159THEN180
252 X2=X2-8:X1=X1+2:X=X-40:POKEV,X1:POKEV+1,X2:POKERI,248
253 X2=X2-4:X1=X1+6:X=X+1:POKEV,X1:POKEV+1,X2:IFPEEK(X+40)<30THENGOSUB280
254 IFPEEK(X)=38THENGOSUB290
255 X2=X2+4:X1=X1+6:X=X+1:POKEV,X1:POKEV+1,X2:POKERI,247
256 IFPEEK(X+40)>159THENX1=X1+2:POKEV,X1:GOTO160
257 X2=X2+8:X1=X1+2:X=X+40:POKEV,X1:POKEV+1,X2
258 IFPEEK(X+40)>159THEN160
259 X2=X2+8:POKEV+1,X2:X=X+40:IFPEEK(X+40)>159THEN160
260 GOTO300
280 Z=PEEK(X+40):POKEX+40,27
285 FORI=1TO10*RU:SYS36864:SC=SC+1:POKES+4,32:POKES+1,IAND255:POKES+4,33:NEXT
286 :IFZ=29THENPOKEX+40,29:RETURN
287 POKEX+40,32:RETURN
290 BS=BS+1:FORI=1TO10*BS*RU:SYS36864:SC=SC+1:POKES+4,32:POKES+1,IAND255
297 POKES+4,33:NEXT:POKEX,32:RETURN
300 X3=PEEK(RI):POKES+4,33
310 FORI=X2TO255:POKEV+1,I:POKES+1,255-I:M=1-M:POKERI,X3+M:NEXT
320 GN=GN-1:IFGN=0THEN500
330 POKE1701,47+GN:X=1947:FORI=1TO5:POKEX+I,32:NEXT
340 POKEV,37:POKEV+1,222:POKERI,248:GOTO160
400 GOSUB800:POKES+1,0:POKES+4,33:FORI=BOTO0STEP-1:BO=BO-1:SC=SC+1
405 SYS36864:SYS36886:NEXT:T=3:GOSUB700
410 RU=RU+1:BO=500*RU:X=1947:BS=0:GOSUB5600:IFRU=4THENGN=GN+1
420 PRINT"[illegible]"TAB(9)"GET READY FOR ROUND"RU:T=4:GOSUB700
430 GOSUB5500:GOSUB5600:GOSUB40000
440 POKE1701,47+GN:POKE1781,48+RU
450 GOTO340
```

```
500 FORI=1493TO2013STEP40:POKEI,39:POKEI+S,9:NEXT
510 POKEV,245:FORI=254TO118STEP-4:POKEV+1,I:M=1-M:POKERI,249+M
520 FORII=1TO80:NEXTII,I:POKERI,245
530 PRINT"■■■■■■■■■■■■"TAB(17)"■GAME OVER"
540 FORI=14TO1STEP-.5:POKE1439+S+I,1:POKEV,PEEK(V)-4:POKERI,245+M:M=1-M
550 FORII=1TO80:NEXTII,I
555 PRINT"■■■■■■■■■■"SPC(15)"■■PQRS■■■■■TUVW"
560 PRINT"■■■■■"TAB(13)"■PRESS FIRE!
565 GOSUB46000:GOTO5000
600 IFBS=10THENGOSUB1000:POKE1086,61
610 RETURN
700 FORI=1TOT:FORJ=1TO700:NEXTJ,I
710 RETURN
800 POKES+4,129:POKES+1,250:T=2:GOSUB700
805 POKES+4,0:POKES+1,90:GOSUB700
810 FORI=1069TO1189STEP40:IFPEEK(I)=29THENS1=I+40:GOTO830
820 NEXT
825 S1=1069
830 FORI=1288TO1448STEP40:IFPEEK(I)=29THENS2=I+40:GOTO850
840 NEXT
850 S3=S1:GOSUB880:S3=S2:GOSUB880
870 RETURN
880 POKES3-40,32:FORI=S3TO2023STEP40:J=PEEK(I):POKEI,29:POKEI+S,5:POKES+4,33
890 FORK=1TO90:NEXT:POKEI,J:POKES+4,32:NEXT:RETURN
1000 FORI=1888TO1892:POKEI,32:POKEI+40,32:POKEI+80,32:NEXT
1010 FORI=1810TO1970STEP40:POKEI,39:POKEI+S,9:NEXT:RETURN
5000 GOSUB5500:IFSC$>=HI$THENHI$=SC$
5030 POKEV+32,2:POKEV+33,2:PRINT"■■■■■■■■■■■■■"TAB(9)"■PRESS FIRE TO START !"
5035 POKEV,0:T=1:GOSUB700:GOSUB46000
5040 X=1947:BS=0:RU=1:BO=500:GN=3:SC=0
5045 GOSUB5600:GOSUB5500:POKEV+1,222:POKEV,37:POKERI,248
5050 POKEV+2,210:POKEV+3,222:POKEV+17,1
5100 GOSUB40000:GOTO160
5500 SC$="■■"+RIGHT$("000000"+MID$(STR$(SC),2),6):RETURN
5600 BO$="■■"+RIGHT$("0000"+MID$(STR$(BO),2),4):RETURN
20000 REM****************
20010 REM* INSTRUCTIONS *
20020 REM****************
20030 POKEV+21,0:PRINT"■":POKE53281,12:POKE53280,12:GOSUB21000
20040 PRINT"■"TAB(13)"■INSTRUCTIONS■":GOSUB21000:POKERI,245
20050 PRINT"■■THIS IS YOUR GNOME■":POKEV+21,1:POKEV,240:POKEV+1,99:GOSUB21000
20060 PRINT"■■JUMP OVER BAD SPIDERS   ■■  ■10 PT * ROUND"
20070 PRINT"■■JUMP OVER  1ST KEY      ■■  ■10 PT * ROUND"
20080 PRINT"■■JUMP OVER 10TH KEY      ■■ ■100 PT * ROUND"
20090 PRINT"■■THE DIRTY SPIDER        ■■ ■(DO NOT TOUCH)■":GOSUB21000
20100 PRINT"■■FIND SPRAY TO KILL SPIDERS■":POKEV+21,3:POKEV+3,210:POKEV+2,240
20110 GOSUB21000:GOSUB46000:GOTO5000
21000 POKES+4,32:POKES+1,15:POKES+5,0:POKES+6,240
21010 FORI=1TO40:PRINT"■-";:POKES+4,33:FORII=1TO5:POKES+4,32:NEXTII,I:RETURN
25000 REM*****************
25010 REM* MUSIC SET     *
25020 REM*****************
25030 S=54272:FORI=0TO24:POKES+I,0:NEXT
25040 POKES+24,79:POKES+5,10:POKES+6,10
25050 FORI=0TO31:READA:FORJ=0TO7:SO=.5-SO:POKE28672+I+32*J,A*(1+SO)
25060 NEXTJ,I
25100 DATA20,25,30,0,10,0,10,0,25,30,35,0,10,0,10,0,20,25,30,25,20,25,30,25,20
25105 DATA25,30,0,15,0,15,0
25900 GOTO30000
30000 REM*****************
30010 REM*NEW CHARACTERS*
```

```
30020 REM*****************
30030 READB
30040 A=PEEK(63):IFA=200THEN35000
30050 FORI=0TO7:POKE14336+I+A*8,B
30060 READB:Q=Q+B:NEXT:GOTO30040
32027 DATA0,0,0,0,0,195,90,165
32028 DATA0,0,0,195,60,90,165,129
32029 DATA129,102,24,195,60,90,165,129
32030 DATA16,8,16,8,16,8,16,8
32031 DATA60,0,0,0,0,60,0,0
32038 DATA60,66,66,60,8,8,56,24
32039 DATA195,195,255,195,195,195,255,195
32045 DATA0,60,60,126,60,24,0,0
32048 DATA60,102,102,102,102,102,60,0
32060 DATA0,126,102,102,102,126,0,0
32061 DATA0,126,126,126,126,126;0,0
32087 DATA24,24,56,240,240,56,24,24
32108 DATA0,0,0,24,24,24,24,24
32126 DATA24,24,24,24,24,0,0,0
32127 DATA60,66,153,161,161,153,66,60
32128 DATA240,248,252,254,255,255,255,249
32129 DATA7,15,31,63,127,255,255,207
32130 DATA159,159,159,159,158,158,158,159
32131 DATA252,254,255,255,31,15,31,255
32132 DATA240,240,240,248,248,252,252,254
32133 DATA135,7,7,15,15,31,31,63
32134 DATA159,159,159,159,158,158,159,159
32135 DATA255,254,252,30,15,15,31,31
32136 DATA192,112,40,38,17,8,9,5
32137 DATA16,32,32,64,64,161,25,6
32138 DATA25,33,65,130,130,4,4,4
32139 DATA30,225,0,0,0,1,6,24
32140 DATA5,8,144,160,64,224,17,9
32141 DATA136,104,16,32,64,128,0,0
32142 DATA224,0,0,1,6,24,96,128
32143 DATA10,12,48,192,0,0,0,0
32144 DATA0,0,3,14,29,29,62,63
32145 DATA15,255,255,59,219,251,120,187
32146 DATA255,255,255,182,183,183,55,183
32147 DATA0,248,252,14,191,191,191,191
32148 DATA61,30,15,7,0,0,0,0
32149 DATA187,123,255,255,15,0,0,1
32150 DATA183,183,255,255,252,224,192,128
32151 DATA190,188,248,224,0,0,0,0
32154 DATA255,0,255,0,255,0,255,0
32155 DATA252,0,252,0,252,0,252,0
32156 DATA128,0,224,0,248,0,254,0
32157 DATA254,0,248,0,224,0,128,0
32158 DATA127,0,31,0,7,0,1,0
32159 DATA1,0,7,0,31,0,127,0
32160 DATA255,192,48,12,3,255,0,0
32161 DATA255,3,12,48,192,255,0,0
32163 DATA223,223,223,0,251,251,251,0
32164 DATA31,31,31,0,3,3,3,0
32165 DATA192,192,192,0,248,248,248,0
32166 DATA255,131,140,176,192,193,177,141
32167 DATA131,3,13,49,193,255,0,0
32200 DATA1
35000 REM*****************
35010 REM* READ SPRITES *
35020 REM*****************
```

COMPUTERCLUB 64

Wien – Ein Verein stellt sich vor

Der *COMPUTERCLUB 64* ist ein seit Mai 1983 eingetragener Verein zur Verwendung und Benutzung des Computers. Seit der Gründung richten sich die Aktivitäten auf den Commodore 64 und allen dazu kompatiblen Geräten.

Aufgaben und Aktivitäten

Der ‚COMPUTERCLUB 64' besitzt eine vereinseigene Buchbibliothek, ein Eprom-Programmiergerät mit einem UV-Löschgerät, Originalprogramme mit Anleitungen und sonstiges notwendiges Zubehör.

Im monatlich erscheinenden Mitteilungsblatt ‚CLUB 64' werden interessante Artikel veröffentlicht, Neuerscheinungen vorgestellt, Programme beschrieben, Tips und Tricks erklärt u. v. m. Alle Mitglieder haben die Möglichkeit, aktiv an der Gestaltung mitzuarbeiten. Darüber hinaus können sie darin kostenlos Privatinserate aufgeben.

Um einen engeren Kontakt zwischen unseren Mitgliedern zu erreichen, werden regelmäßig Vereinstreffen – vorerst nur in Wien – abgehalten. Diese Meetings erfreuen sich großer Beliebtheit und sind überdurchschnittlich besucht. Dabei werden Probleme ausdiskutiert und gelöst, eigene Programm- und Hardware-Entwicklungen vorgestellt, Sammelbestellungen organisiert; dem Einfallsreichtum und der Kreativität der Mitglieder sind dabei keine Grenzen gesetzt.

Besonders interessant sind die einzelnen Sektionen – User Groups genannt. Zur Zeit gibt es drei solcher, die sich mit PASCAL, COMAL und MASCHINENSPRACHE befassen. Zweck und Ziel dieser Sektionen ist ein reger Informations- und Erfahrungsaustausch sowie die Übermittlung von fundiertem Wissen.

Um jedoch den gesellschaftlichen Sinn nicht zu vernachlässigen, werden auch viele Freizeitaktivitäten zum Selbstkostenpreis angeboten. Vom Sommernachtsfest, einer abenteuerlichen Rätselrallye bis hin zu einem Raubrittermahl erstreckt sich das breite Angebot. Aus diesen Gründen herrscht ein äußerst freundschaftliches Vereinsleben, bei dem jedoch nicht auch die ‚Computerei' vergessen wird. Bei Wünschen, Anregungen und Anmeldungen wenden Sie sich bitte an: Mag. Helmut Hackl, 1030 Wien, Erdbergstraße 134/4/7, Tel. 02 22 / 73-87-722; 1010 Wien.

```
35030 FORI=0TO6:FORJ=0TO62:READA
35040 POKE15680+64*I+J,A:NEXTJ,I
35050 REM****************
35060 REM* SPRITE RIGHT *
35070 REM****************
35080 DATA0,1,0,0,5,0,0,5,64,0,21,64,0,63,64,0,255,64,0,63,0,0,15,0,0,41,64
35090 DATA3,105,64,3,85,128,0,166,64,0,63,192,0,42,128,0,42,64,0,25,128
35100 DATA0,26,80,0,10,80,0,5,80,0,5,16,0,21,0
35110 DATA0,1,0,0,5,0,0,5,64,0,21,64,0,63,64,0,255,64,0,63,0,0,15,0,0,41,64
35120 DATA3,105,64,3,85,128,0,166,64,0,63,192,0,106,128,0,170,64,0,162,128
35130 DATA0,162,128,0,82,128,0,81,64,1,81,64,0,5,64
35140 REM****************
35150 REM* SPRITE LEFT  *
35160 REM****************
35170 DATA0,4,0,0,5,0,0,21,0,0,21,64,0,31,192,0,31,240,0,15,192,0,15,0,0,22,128
35180 DATA0,22,156,0,37,92,0,25,160,0,63,192,0,42,128,0,26,128,0,38,64,0,90,64
35190 DATA0,90,0,0,85,0,0,69,0,0,5,64
35200 DATA0,4,0,0,5,0,0,21,0,0,21,64,0,31,192,0,31,240,0,15,192,0,15,0,0,22,128
35210 DATA0,22,156,0,37,92,0,25,160,0,63,192,0,26,144,0,26,144,0,42,160
35220 DATA0,40,160,0,40,160,0,40,80,0,20,84,0,21,0
35230 REM****************
35240 REM* SPRITE UP/DO *
35250 REM****************
35260 DATA0,1,0,0,5,0,0,5,64,0,21,64,0,21,64,0,37,192,0,31,64,0,15,0,0,42,128
35270 DATA0,106,144,0,106,144,0,106,144,0,223,112,0,218,176,0,42,144,0,42,160
35280 DATA0,40,160,0,40,80,0,40,80,0,20,0,0,20,0
35290 DATA0,1,0,0,5,0,0,5,64,0,21,64,0,21,64,0,37,192,0,31,64,0,15,0,0,42,128
35300 DATA0,106,144,0,106,144,0,106,144,0,223,112,0,218,176,0,42,144,0,42,160
35310 DATA0,40,160,0,20,160,0,20,160,0,0,80,0,0,80
35320 REM****************
35330 REM* SPRITE SPRAY *
35340 REM****************
35350 DATA0,0,0,0,0,0,0,0,0,0,0,0,28,0,0,20,0,0,28,0,0,8,0,0,127,0,0,255,128
35360 DATA0,255,128,0,193,128,0,128,128,0,182,128,0,148,128,0,193,128,0,213,128
35370 DATA0,221,128,0,227,128,0,255,128,0,0,0
36000 REM****************
36010 REM*MASCHINE CODES*
36020 REM****************
36030 FORI=36864TO37243:READA:POKEI,A:NEXT:GOTO45000
36040 DATA162,5,189,96,5,201,57,240,4,254,96,5,96,169,48,157,96,5,202,76,2,144
36050 DATA162,3,189,40,6,201,48,208,9,224,0,240,4,202,76,24,144,96,162,3,189,40
36060 DATA6,201,48,240,4,222,40,6,96,169,57,157,40,6,202,76,42,144,234,234,234
36070 DATA234,234,234,162,0,160,1,173,183,4,141,255,143,185,183,4,157,183,4,232
36080 DATA200,224,6,208,244,162,0,160,1,173,155,7,141,188,4,185,155,7,157,155,7
36090 DATA232,200,224,13,208,244,162,0,160,1,173,203,4,141,167,7,185,203,4,157
36100 DATA203,4,232,200,224,18,208,244,173,255,143,141,220,4,96,162,0,160,1,173
36110 DATA253,4,141,255,143,185,253,4,157,253,4,232,200,224,5,208,244,162,0,160
36120 DATA1,173,202,5,141,1,5,185,202,5,157,202,5,232,200,224,5,208,244,162,0
36130 DATA160,1,173,39,7,141,206,5,185,39,7,157,39,7,232,200,224,5,208,244,173
36140 DATA255,143,141,43,7,96,160,0,232,200,192,40,208,250,96,160,0,202,200,-192
36150 DATA40,208,250,96,165,253,201,0,208,39,173,168,5,201,29,208,5,162,1,134
36160 DATA253,96,162,0,189,8,5,201,29,208,12,254,8,5,169,29,32,214,144,157,8,5
36170 DATA96,32,214,144,76,252,144,173,8,5,201,29,208,5,162,,134,253,96,162,160
36180 DATA189,8,5,201,29,208,18,254,8,5,254,8,5,254,8,5,169,29,32,223,144,157,8
36190 DATA5,96,32,214,144,76,35,145,162,32,142,4,212,174,288,142,189,,112,141,1
36200 DATA212,238,238,142,162,33,142,4,212,96,162,160,189,8,5,157,5,4,32,223
36210 DATA144,224,,208,243,96,32,22,144,32,68,144,32,141,144,32,232,144,32,66
36220 DATA145,32,89,145,96
40000 REM****************
40010 REM* GAME-SCREEN  *
40020 REM****************
```

```
40030 POKEV+32,9:POKEV+21,3:POKEV+33,0
40060 PRINT"###############################################";
40070 PRINT"###HIJ "SPC(16)"###%HIGH :##";
40080 PRINT"###KLM "SPC(16)"###%------##";
40090 PRINT"###NO  "SPC(16)"###%"HI$"##";
40100 PRINT"##%       £ ############";
40110 PRINT"##% £                  ! ! ! ############";
40120 PRINT"## !  ! ! ! !";
40122 IFRU<2THENPRINT" ! ! ";:GOTO40125
40123 IFRU<4THENPRINT" ! !←";:GOTO40125
40124 PRINT" !←←←";
40125 PRINT"! !  ]     ##%SCORE:##";
40130 PRINT"##%"SPC(19)"     ##%------##";
40140 PRINT"##%&"SPC(16)"     ##%"SC$"##";
40150 PRINT"##%"SPC(13)" &   ############";
40160 PRINT"## ! ! ! "SPC(13)"    ############";
40170 PRINT"##%  !←←←←! !  ! ##%BONUS:##";
40180 PRINT"##%  "SPC(10)"  ##%----   ##";
40190 PRINT"##%   ##%"BO$"  ###";
40200 PRINT"## ! ! ! ! "SPC(14)" ############";
40210 PRINT"##%"SPC(14)"& ! ! ! !############";
40220 PRINT"##%"SPC(12)"&   ##%GNOME:2##";
40230 PRINT"##%"SPC(9)" !   ##############";
40240 PRINT"##%&  "SPC(11)" ##%ROUND:1##";
40250 PRINT"##%"SPC(9)" "SPC(9)"$################";
40260 PRINT"##! ! ! ! ! ! ! ############%  1984##";
40270 PRINT"##%"SPC(17)"$###########%     ##";
40280 PRINT"##%  ###############%  @ABC ##";
40290 PRINT"##%        ###############%  DEFG ##";
40300 PRINT"################################################"CHR$(148)"##"
40900 POKEV+17,27
40999 RETURN
45000 REM****************
45010 REM* TITEL        *
45020 REM****************
45030 POKEV+32,4:POKEV+33,4
45040 PRINT"";:FORI=1TO8:PRINT"@ABC";:NEXT
45050 FORI=1TO8:PRINT"DEFG";:NEXT
45060 PRINT"SOFT ";:FORI=1TO5:PRINT"- SOFT ";:NEXT
45090 POKEV,230:POKEV+1,180:POKERI,245:POKEV+21,1
45100 PRINT:PRINT
45110 PRINTTAB(8)"
45120 PRINTTAB(8)"
45130 PRINTTAB(8)"
45140 PRINT"←ZZZ£ZZZZ£[ZZZZ£ZZZZ[ZZZZ£←ZZZ£"
45150 PRINT"Z]↑]Z↑[[Z↑[ZZ↑[Z]↑]"
45160 PRINT"ZZ[[Z[ZZ[Z"
45170 PRINT"Z£Z←[[Z[ZZ←[Z£"
45180 PRINT"↑ZZZ£ZZZZ][Z[ZZZZZZZ]↑ZZZ£"
45190 PRINT"↑[Z[Z[ZZ↑£↑Z"
45200 PRINT"[Z[Z[ZZ[Z"
45210 PRINT"←£←[Z[Z←[ZZ[←£←Z"
45220 PRINT"↑ZZZ]Z £[ZZZZ]ZZZZ[Z[ ↑ZZZ]"
45230 PRINT" 1984  BY  MR←SOFT  RONALD MAYER"
45235 PRINT"";:FORI=1TO8:PRINT"@ABC";:NEXT
45237 FORI=1TO8:PRINT"DEFG";:NEXT:GOSUB46000
45240 FORI=230TO115STEP-3:FORJ=1TO50:NEXTJ:SB=1-SB:POKE2040,245+SB:POKEV,I:NEXT
45250 GOTO20000
46000 O=O+1:IFO=4THENSYS37186:O=0
46010 IFPEEK(56320)=111THENPOKES+1,0:RETURN
46020 GOTO46000
```

Dragonpaint



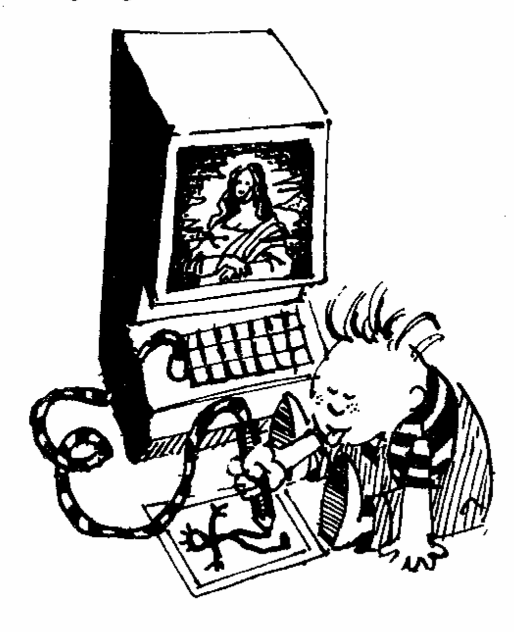
Dragonpaint ist ein menügesteuertes Basicprogramm zur einfachen Erstellung von Bildschirmgrafiken auf dem Dragon 32 und dem Dragon 64.
Die Grafiken werden mit Hilfe eines Joysticks und der Tastatur erstellt.

Sie können auch auf einem Drucker ausgegeben werden.
Mit dem rechten Joystick wird der Zeichenstift über den Bildschirm gesteuert. Die Position des Zeichenstiftes ist die, ab der gezeichnet, angemalt oder gelöscht wird. Bei Kreisen oder Ellipsen gibt der Stift den Mittelpunkt und den Radius an.

```
10 REM************
20 REM*  d-Paint  *
30 REM*(C) 1984   *
40 REM*VOLKER     *
50 REM*MARKMILLER*
60 REM************
70 REM***einleitung***
80 CLS:PRINT@70,"ICH BEGRUESSE SIE ZU DEM"
90 PRINT@134,"ZEICHENPROGRAMM"
100 PRINT@263,"dragonPaint"
110 FOR A=1 TO 3000:NEXT A
120 REM***sPeedPoke***
130 CLS:PRINT"KANN IHR DRAGON SPEEDPOKE?"
140 PRINT "(J/N)
150 INPUT A$
160 IF A$="J" THEN GOTO 190
170 IF A$="N" THEN GOTO 200
180 GOTO 150
190 POKE &HFFD7,1
200 PMODE 4,1:SCREEN1,1:PCLS
210 CLS:PRINT" ES FOLGT NUN EINE LISTE MIT"
220 PRINT" DEN ANWEISUNGEN DIE SIE UEBER"
230 PRINT" DIE TASTATUR ABRUFEN KOENNEN."
240 PRINT" SIE HAT DIE GLEICHE REIHEN-"
250 PRINT" FOLGE WIE DIE AUF DEM"
260 PRINT" GRAFIKBILDSCHIRM DARGESTELLTEN"
270 PRINT" BEFEHLE."
280 PRINT@416, "WEITER DANN --> W"
290 IF INKEY$="W" THEN GOTO 310
300 GOTO 290
310 CLS:REM***befehlsliste***
320 PRINT@0 ,"l LINIE":PRINT@32,"k KREIS":PRINT@ 64,"z ZEICHNE MIT DEN CURSORTAS
TEN":PRINT@ 96,"a ANMAHLEN":PRINT@128,"u UEBERMALEN (LOESCHEN)":PRINT@160,"e KRE
ISBOGEN":PRINT@192,"@ ZURUECK ZUR BEFEHLSLISTE"
330 PRINT@448,"WEITER--->W"
340 IF INKEY$="W" THEN GOTO 360
350 GOTO 340
360 CLS:PRINT@  32,"s ZURUECK IN DAS STEUERPROGRAMM":PRINT@ 64,"0 BILDSCHIRM LOE
SCHEN":PRINT@ 96,"h BILDSCHIRMAUSDRUCK":PRINT@128,"f PFEILPOSITION ABFRAGEN":PRI
NT@160,". PFEILPOSITION ZEIGEN":PRINT@192,"b RECHTECK"
370 PRINT @224,"; PUNKTIEREN"
380 PRINT@256,": FLAECHEN ANMALEN"
390 PRINT@288,"c BILD ABSPEICHERN":PRINT@448,"WEITER ---> W"
400 IF INKEY$="W" THEN GOTO 430
410 S1=PEEK(341):IF S1=239 THEN GOTO 870
420 GOTO 400
430 CLS:PRINT"WOLLEN SIE EIN GITTER? (J/N)
440 INPUT A$
450 IF A$="J" THEN GOTO480
460 IF A$="N" THEN GOTO 580
470 GOTO 440
480 CLS: PRINT "WELCHE GITTERGROESSE WOLLEN SIE?
490 INPUT A
500 PMODE4,1:SCREEN1,1
510 FOR Z=0 TO255 STEP A
520 LINE (Z,0)-(Z,188),PSET
530 NEXT Z
540 FOR Z=0 TO 188 STEP A
550 LINE (0,Z)-(255,Z),PSET
560 NEXT Z
570 REM***maske***
580 PMODE 4,1:SCREEN1,1
590 DRAW"BM2,4 D20R252U20L252"
600 DRAW"BM2,4R25D20R25U20R25D20R25U20R25D20R25U20R25D20R25U20R25D20R25U20"
610 DRAW"BM6,20E10BM19,13U6BD6R4BL5"
620 CIRCLE (35,12),4,5:DRAW"BM45,12U6BD3R1E3BG3F3BL4"
630 DRAW "BM56,20R5U5L4U3R7U4BM68,15BU6R4D1G4D1R4BL4"
640 LINE (80,8)-(85,12),PSET,BF:DRAW"BM92,15U4E2F2D4BU2L4BD2"
650 LINE (105,8)-(114,17),PSET,B
```

```
660 A=105:B=8
670 LINE (105,8)-(114,B),PSET
680 B=B+1
690 IF B=17 THEN GOTO 710
700 GOTO 670
710 DRAW"BM 120,15 BU6D5F1R2E1U5BD6BL4"
720 CIRCLE (135,12),6,,0.6:DRAW"BM144,15 U6R4BG3BL1R3BG3R4BL4"
730 LINE (155,8)-(165,18),PSET,B:DRAW"BM158,10NR4BD2NR4BD2NR4BD2R4":DRAW"BM170,1
5BR2BU2U1L1G1D1F1R2E1U4H1L2G1BD5"
740 DRAW"BM185,22U14NF4G4":DRAW"BM195,15R3E1U1H1L2H1U1E1R3BD6BL4"
750 LINE (205,8)-(215,18),PSET,B:DRAW"BM220,15BU1U4E1R2F1D4G1L2H1BD1"
760 DRAW"BM233,20 R10U5L10R2U5R6D5L8D5U3L3U2D4U2R3D3R10U3R3U2D4"
770 DRAW"BM244,12U6BD3RR4BU3D6BL4"
780 DRAW "BM254,24L20D80R20U80D20L20D20R20D20L20"
790 DRAW"BM246,41U6R3F1D1G1L3BR3F1D1G1L3":LINE (242,27)-(250,32),PSET,B
800 LINE (242,47)-(248,52),PSET,BF:DRAW"BM246,61BR2U1BU2U1BD4BL2"
810 FOR R=1 TO9
820 A=RND (6):B=RND(6)
830 PSET (242+A,67+B,5):PSET (242+B,67+A,5)
840 NEXT R
850 DRAW"BM 247,81BR1EE1U1BU2U1BD4BL2"
860 DRAW"BM 240,86R9D5R1NL11D2L11U2R1U5":DRAW"BM 242,102BU1U4E1R2F1BD4G1L2H1BD1"
870 PMODE4,1:SCREEN1,1:REM***vereinbahrungen***
880 PMODE4,1:SCREEN1,1:S1=PEEK(341):Z1=PEEK(340):L1=PEEK(342):K1=PEEK(341):A1=PE
EK(339):H1=PEEK(338):U1=PEEK(343):F1=PEEK(344):PU1=PEEK(344):AA1=PEEK(338):E1=PE
EK(343):NU=PEEK(338):B1=PEEK(340):ST=PEEK(341):D1=PEEK(340):C1=PEEK(341)
890 REM***abfrage und verteiler**
900 A=JOYSTK(0):B=JOYSTK(1):A=A*4:B=B*3
910 LINE (A,B)-(A+ 5,B+ 5),PSET:LINE (A,B)-(A+5,B),PSET:LINE (A,B)-(A,B+5),PSET
920 LINE (A,B)-(A+ 5,B+ 5),PRESET:LINE (A,B)-(A+5,B),PRESET:LINE(A,B)-(A,B+5),PR
ESET
 930 IF Z1=223 THEN GOTO 1330
 940 IF L1=247 THEN GOTO 1120
 950 IF K1=247 THEN GOTO 1210
 960 IF A1=251 THEN GOTO 1560
 970 IF H1=247 THEN GOTO 1900
 980 IF U1=239 THEN GOTO 1420
 990 IF F1=251 THEN GOTO 1830
1000 IF PU1=253 THEN GOTO 1680
1010 IF AA1=251 THEN GOTO 310
1020 IF E1=251 THEN GOTO 2670
1030 IF NU=254 THEN GOTO 1100
1040 IF B1=251 THEN GOTO 2960
1050 IF ST=253 THEN GOTO 3290
1060 IF D1=253 THEN GOTO 3260
1070 IF C1=251 THEN GOTO 3410
1080 GOTO 880
1090 REM***alles loeschen***
1100 PCLS : GOTO 570
1110 REM***linie***
1120 X=JOYSTK(0):Y=JOYSTK(1):X=X*4:Y=Y*3
1130 LINE (A,B)-(X,Y),PSET
1140 IF X<>V OR Y<>O THEN LINE (A,B)-(X,Y),PRESET
1150 IF X=V OR Y=O THEN LINE (A,B)-(X,Y),PRESET
1160 S1=PEEK(341):IF S1=239 THEN GOTO 1190
1170 X=V:Y=O
1180 GOTO 1120
1190 LINE (A,B)-(X,Y),PSET:GOTO 880
1200 REM***kreis***
1210 V=JOYSTK(0):C=PEEK(65280):V=V*4
1220 LINE (V,B)-(V+ 5,B+ 5),PSET:LINE (V,B)-(V+5,B),PSET:LINE (V,B)-(V,B+5),PSET
1230 LINE (V,B)-(V+ 5,B+ 5),PRESET:LINE (V,B)-(V+5,B),PRESET:LINE (V,B)-(V,B+5),
```

Die einzelnen Tastaturbefehle:
>L< Es wird ab der aktuellen Stiftposition mit dem Joystick eine beliebige Linie gezeichnet. Wenn Sie gesetzt werden soll dann >S<.
>Z< Mit den Cursortasten wird eine beliebige Linie in die Cursorpfeilrichtungen gezeichnet.



>K< Der Stift wird auf den Kreismittelpunkt gesetzt, dann K. Dann wird der Stift auf den Kreisrand gesetzt und der Feuerknopf gedrückt.
>A< Mit den Cursortasten kann man eine Fläche ausmalen.
>U< Mit den Cursortasten kann man Linien oder Flächen löschen.
>E< Nach dem gleichen Prinzip wie beim Kreis lassen sich nach Eingabe der Exzentrität Elypsen oder Bögen zeichnen.
>B< Mit B läßt sich der ganze Bildschirm löschen.
>H< H gibt eine Hardcopy auf einen Drucker aus.
>F< Gibt die aktuelle Stiftposition an.
>.< Mit . kann man sich eine bestimmte Stiftposition zeigen lassen.
>O< Springt in das Befehlsmenü zurück so oft man will.
>S< Springt immer in das Steuerprogramm zurück. Nur von diesem Programm aus kann man neue Befehle erteilen.
Es ist wichtig, daß man den Joystick langsam bewegt und daß man, wenn man z. B. über den ganzen Bildschirm wandern muß, möglichst am Rand entlanggeht.
Die Hardcopy dauert etwas, da erst das Bildschirmmenü weggeplottet wird, damit es nicht auf den Ausdruck kommt.
Speedpoke können nur die Dragon 32

```
PRESET
1240 PSET (A,B,5)
1250 R=V-A
1260 IF R<0 THEN R=R*(-1)
1270 S1=PEEK(341):IF S1=239 THEN GOTO 880
1280 IF C=126 OR C=254 THEN GOTO1300
1290 GOTO 1210
1300 CIRCLE  (A,B),R,5
1310 GOTO 1210
1320 REM***zeichnen***
1330 F=A:G=B
1340 A=PEEK(341):B=PEEK(342):C=PEEK(343):D=PEEK(344)
1350 IF A=223 THEN G=G-1
1360 IF B=223 THEN G=G+1
1370 IF C=223 THEN F=F-1
1380 IF D=223 THEN F=F+1
1390 PSET (F,G,5)
1400 S1=PEEK(341):IF S1=239 THEN GOTO 880
1410 GOTO 1340
1420 F=A:G=B:REM***loeschen***
1430 A=PEEK(341):B=PEEK(342):C=PEEK(343):D=PEEK(344)
1440 IF A=223 THEN G=G-3
1450 IF B=223 THEN G=G+3
1460 IF C=223 THEN F=F-3
1470 IF D=223 THEN F=F+3
1480 FOR Q=1 TO 3
1490 CIRCLE (F,G),Q,0
1500 NEXT Q
1510 CIRCLE (F,G),3,5
1520 CIRCLE (F,G),3,0
1530 S1=PEEK(341):IF S1=239 THEN GOTO 870
1540 GOTO 1430
1550 REM***anmahlen***
1560 F=A:G=B
1570 GOTO 1630
1580 A=PEEK(341):B=PEEK(342):C=PEEK(343):D=PEEK(344)
1590 IF A=223 THEN G=G-1
1600 IF B=223 THEN G=G+1
1610 IF C=223 THEN F=F-1
1620 IF D=223 THEN F=F+1
1630 FOR Q=1 TO 2
1640 CIRCLE (F,G),Q,5
1650 NEXT Q
1660 S1=PEEK(341):IF S1=239 THEN GOTO 870
1670 GOTO 1580
1680 REM***Pos.zeigen***
1690 CLS:PRINT" WELCHE BILDSCHIRM POSITION"
1700 PRINT "MOECHTEN SIE SEHEN?"
1710 INPUT A
1720 INPUT B
1730 PMODE 4,1:SCREEN1,1
1740 FOR F=1 TO5
1750 PSET (A+F,B,5)
1760 NEXT F
1770 S1=PEEK(341):IF S1=239 THEN GOTO 1790
1780 GOTO 1740
1790 FOR F=1 TO 5
1800 PRESET (A+F,B,0):
```




```
1810 NEXT F
1820 GOTO 870
1830 REM***Pos.abfragen***
1840 CLS:PRINT"DIE PFEILPOSITION IST:"
1850 PRINT A,B
1860 S1=PEEK(341):IF S1=239 THEN GOTO 1880
1870 GOTO 1860
1880 PMODE 4,1:SCREEN1,1:GOTO 870
1890 REM***menue loeschen***
1900 A=0
1910 FOR N=1 TO 255
1920 PSET (N,A,0)
1930 NEXT N
1940 A=A+1
1950 IF A= 25 THEN GOTO 1970
1960 GOTO 1910
1970 A=225
1980 FOR N=0 TO 110
1990 PSET (A,N,0)
2000 NEXT N
2010 A=A+1
2020 IF A=255 THEN GOTO 2040
2030 GOTO 1980
2040 POKE &HFFD6,0:REM***hardcopy***
2050 CLEAR 100,32498
2060 DEF USR0=32500
2070 PRINT#-2,CHR$(27);"A";CHR$(8);
2080 FOR I=32500 TO 32593
2090 READ INH:POKE I,INH
2100 NEXTI
2110 GOSUB 2610
2120 X=USR0(0)
2130 PRINT#-2,STRING$(5,CHR$(13))
2140 CLS:PRINT@200,"ENDE DER AUSGABE"
2150 GOTO 570
2160 REM**** MASCHINENCODE****
2170 DATA &H34,&H7E
2180 DATA &H32,&H75
2190 DATA &H8E,&H06,&H00
2200 DATA &H8D,&H3E
2210 DATA &H1F,&H12
2220 DATA &H31,&HA8,&H20
2230 DATA &H10,&HAF,&H69
2240 DATA &H1F,&H12
2250 DATA &HC6,&H07
2260 DATA &HE7,&H68
2270 DATA &HA6,&HA4
2280 DATA &HA7,&HE5
2290 DATA &H31,&HA8,&H20
2300 DATA &H5A
2310 DATA &H2A,&HF6
2320 DATA &HC6,&H07
2330 DATA &H69,&HE5
2340 DATA &H49
2350 DATA &H5A
2360 DATA &H2A,&HFA
2370 DATA &H34,&H30
2380 DATA &HBD,&H80,&H0F
2390 DATA &H35,&H30
2400 DATA &H6A,&H68
2410 DATA &H2A,&HED
2420 DATA &H30,&H01
```

der ersten Serie und die Dragon 64 im 32k Modus.
Zum Ausprobieren: der Poke ist &HFFD 7,1

Zusatzanleitung zum Programm Dragonpaint

>B< Man setzt den Zeichenpfeil an die Stelle, wo man die linke untere Ecke des Rechteckes haben will, dann B, und erzeugt mit den Joystick ein beliebiges Rechteck.

>:< Man setzt den Zeichenpfeil knapp (ca. 5 Bildpunkte) unter die anzumalende Fläche und drückt :.
Achtung: Die Fläche muß allseitig GANZ umschlossen sein.

>;< Man drückt ; und kann nun mit den Cursortasten eine Fläche punktieren.

>C< Wenn man ein Bild auf Kassette abspeichern will, drückt man C, wartet bis die Maske gelöscht ist und handelt dann nach dem auftauchenden Menü.

Wenn man ein Bild wieder einladen will, geht man so vor: Man tippt ein: PCLS (ENTER) 10 PMODE 4,1 : SCREEN 1,1 20 GOTO 20. Dann lädt man das Bild mit CLOADM"", R und startet das obige Programm.

```
2430 DATA &HAC,&H69
2440 DATA &H26,&HD7
2450 DATA &H30,&H89,&H00,&HE0
2460 DATA &H8C,&H1E,&H00
2470 DATA &H26,&HC4
2480 DATA &H32,&H6B
2490 DATA &H35,&HFE
2500 DATA &H33,&H8C,&H0F
2510 DATA &HC6,&H05
2520 DATA &HA6,&HC0
2530 DATA &H34,&H54
2540 DATA &HBD,&H80,&H0F
2550 DATA &H35,&H54
2560 DATA &H5A
2570 DATA &H26,&HF4
2580 DATA &H39
2590 DATA &H0A,&H1B,&H4B,&H00
2600 DATA &H01
2610 CLS:PRINT@ 200,"DAS FOLGENDE BILD ";
2620 PRINT @264,"WIRD AUSGEGEBEN";
2630 FOR J=1 TO 500:NEXT J
2640 PMODE 4,1:SCREEN1,1
2650 FOR J=1 TO 1000:NEXT J
2660 RETURN
2670 REM***kreisbogen***
2680 PMODE4,1:SCREEN1,1
2690 V=JOYSTK(0):V=V*4:C=PEEK(65280)
2700 LINE (V,B)-(V+ 5,B+ 5),PSET:LINE (V,B)-(V+5,B),PSET:LINE (V,B)-(V,B+5),PSET
2710 LINE (V,B)-(V+ 5,B+ 5),PRESET:LINE (V,B)-(V+5,B),PRESET:LINE (V,B)-(V,B+5),
PRESET
2720 R=V-A
2730 IF R<0 THEN R=R*(-1)
2740 PSET (A,B,5)
2750 IF C=126 OR C=254 THEN GOTO 2800
2760 FOR Q=1 TO 10 STEP 2
2770 PRESET (V+Q,B,0)
2780 NEXT Q
2790 GOTO 2690
2800 CLS:PRINT"WELCHE EXZENTRITAET SOLL DIE    ELYPSE HABEN? EX.1 IST EIN KREIS"
2810 INPUT Z
2820 PMODE 4,1:SCREEN1,1
2830 CIRCLE (A,B),R,5,Z
2840 CLS:PRINT "NUR EIN BOGEN? (J/N)"
2850 INPUT A$
2860 IF A$="N" THEN GOTO 870
2870 IF A$="J" THEN CLS
2880 CLS:PRINT "NACH UNTEN ODER NACH OBEN? (U/O)"
2890 INPUT A$
2900 IF A$="O" THEN GOTO 2920
2910 IF A$="U" THEN GOTO 2940
2920 PMODE 4,1:SCREEN1,1:CIRCLE (A,B),R,0,Z,0,.5
2930 GOTO 870
2940 PMODE 4,1:SCREEN1,1:CIRCLE (A,B),R,0,Z,0.5,0
2950 GOTO 870
2960 PMODE 4,1:SCREEN1,1
2970 A=JOYSTK(0):A=A*4:B=JOYSTK(1):B=B*3
2980 FOR X=1 TO 10 STEP 2
2990 PSET (A+X,B,5)
3000 NEXT X
3010 FOR X=1 TO 10 STEP 2
3020 PRESET (A+X,B,0)
3030 NEXT X
```




```
3040 IF INKEY$="B" THEN GOTO 3070
3050 GOTO 2960
3060 REM***felder***
3070 CLS:PRINT "WOLLEN SIE LEERE ODER ANGEMAHLTE FELDER? (L/A)
3080 INPUT A$
3090 IF A$="L" THEN GOTO 3120
3100 IF A$="A" THEN GOTO 3170
3110 GOTO 3080
3120 PMODE4,1:SCREEN1,1:V=JOYSTK(0):V=V*4:O=JOYSTK(1):O=O*3
3130 LINE (A,B)-(V,O),PSET,B
3140 LINE (A,B)-(V,O),PRESET,B
3150 S1=PEEK(341):IF S1=239 THEN GOTO 3240
3160 GOTO 3120
3170 PMODE 4,1:SCREEN 1,1
3180 V=JOYSTK(0):V=V*4:O=JOYSTK(1):O=O*3
3190 LINE (A,B)-(V,O),PSET,BF
3200 LINE (A,B)-(V,O),PRESET,BF
3210 IF INKEY$="S" THEN GOTO 3230
3220 GOTO 3180
3230 LINE (A,B)-(V,O),PSET,BF:GOTO 870
3240 LINE (A,B)-(V,O),PSET,B:GOTO 870
3250 REM***gezielt anmalen***
3260 PAINT (A, B-20),5,5
3270 GOTO 870
3280 REM***punktieren***
3290 F=A:G=B
3300 A=PEEK(341):B=PEEK(342):C=PEEK(343):D=PEEK(344)
3310 IF A=223 THEN G=G-3
3320 IF B=223 THEN G=G+3
3330 IF C=223 THEN F=F-3
3340 IF D=223 THEN F=F+3
3350 Y=RND(4):Z=RND(4)
3360 PSET (F+Y,G+Z,5)
3370 PSET (F+Z,G+Y,5)
3380 IF INKEY$="S" THEN GOTO 870
3390 GOTO 3300
3400 REM***menue loeschen***
3410 A=0
3420 FOR N=1 TO 255
3430 PSET(N,A,0)
3440 NEXT N
3450 A=A+1
3460 IF A=25 THEN GOTO 3480
3470 GOTO 3420
3480 A=225
3490 FOR N=0 TO 110
3500 PSET(A,N,0)
3510 NEXT N
3520 A=A+1
3530 IF A= 255 THEN GOTO 3550
3540 GOTO 3490
3550 REM***abspeichern***
3560 POKE&HFFD6,0
3570 CLS:PRINT"BITTE LEGEN SIE EINE LEERE"
3580 PRINT"CASSETTE EIN UND SCHALTEN SIE "
3590 PRINT"AUF AUFNAHME."
3600 PRINT:PRINT:PRINT
3610 PRINT"abspeichern ---> enter"
3620 A=PEEK(338):IF A=191 THEN GOTO 3640
3630 GOTO 3620
3640 CSAVEM"BILD",1536,7679,1536
3650 GOTO 880
```

Cave Man

LASER™ 2001

Schwer hatten es die Menschen in der Steinzeit, denn sie mußten sich ihr Essen bitter erkämpfen. Leo Neandertal ist nun einer dieser Menschen, der seine Familie ernähren muß.

Leo hat die Aufgabe, dem Saurier Rex die Eier zu stehlen, und diese Eier in den unterirdischen Höhlen zu lagern. Freilich hat der Saurier Rex etwas gegen Leo's Unterfangen, und versucht mit allen Mitteln Leo Neandertal am Diebstahl der Eier zu hindern.

Das Spiel ist auf dem Laser 2001 in Laserbasic geschrieben und wird mittels linkem Joystick gesteuert. Es handelt sich um ein reines Basic-Programm, ohne irgend welche Maschinensprache Elemente. Zur Steuerung kann jeder handelsübliche Joystick benutzt werden.

Cave Man zeichnet sich besonders durch die hervorragende High Resolution Grafik aus.

Tippen Sie das Programm ab oder laden Sie dieses von Kassette oder Diskette ein.

Starten Sie das Programm jetzt mit (RUN), und das Titelbild erscheint. Sie haben jetzt die Möglichkeit, zwischen 8 Schwierigkeitsstufen zu wählen, wobei 1 die kleinste Schwierigkeit und 8 die größte Schwierigkeit ist. Tippen Sie einfach die gewünschte Schwierigkeit (Tasten 1 bis 8). Der Rechner schaltet jetzt in den Grafik Modus und zeichnet die Grafik.

Nachdem die Titelschrift „Cave Man" erschienen ist, ertönt eine Fanfare. Das ist das Zeichen, um den linken Aktionsknopf am linken Joystick zu drücken und somit das Spiel zu beginnen.

Leo Neandertal steht links im Bild unter seiner Hütte. Rechts im Bild ist der Saurier Rex zu sehen, welcher immer wieder den Kopf neigt und Feuer speiht. Steuern Sie Leo mittels des Joysticks nach rechts, bis Sie den Saurier Rex erreicht haben. Leo wird sich jetzt automatisch nach links umdrehen, und das Ei des Sauriers erscheint vor Leo. Bewegen Sie den Joystick jetzt nach links, um Leo zusammen mit dem Ei in die Hütte zurückzuführen, damit Leo das Ei in die Höhlen werfen kann.

Sind Sie jedoch im Bereich des Sauriers vorsichtig, da dieser Feuer speiht. Wenn Leo Neandertal von diesem Feuer erwischt wird, so verliert er sein Leben und das Spiel ist zu Ende.

Leo hat jedoch eine Waffe gegen den Saurier Rex: seinen selbstgezimmerten Steinzeithammer. Leo kann diesen Hammer mitnehmen und diesen dem Saurier an den Kopf werfen, wodurch Rex für kurze Zeit betäubt wird. Dann kann Leo ungehindert das Ei stehlen.

Zu Spielbeginn stehen Leo 3 Hämmer zur Verfügung.

Wenn Leo die Absicht hat, sich mit einem Hammer auf den Weg zu Rex zu machen, so muß' er in seiner Hütte stehen, und der Aktionsknopf muß gedrückt werden. Leo wendet sich jetzt dem Saurier zu, und der Hammer erscheint vor Leo. Sobald Leo den Hammer jedoch genommen hat, kann er nur noch Richtung Saurier gehen, aber nicht mehr zur Hütte zurück.

Will Leo jetzt den Hammer werfen, so ist wiederum der Aktionsknopf zu drücken. Es empfiehlt sich jedoch schon während des Fluges des Hammers weiter in Richtung Saurier zu gehen, da die Ohnmacht nur kurz anhält.

Trifft der Hammer den Saurier am Kopf, so sieht der gute Rex jetzt Sternchen. Wenn der Hammer jedoch nicht trifft oder Leo den Saurier erreicht hat, ohne den Hammer zu werfen, so wird ein Hammer abgezogen. Wenn kein Hammer mehr zur Verfügung steht, so ist das Spiel beendet.

Der Saurier Rex ist unserem Leo manchmal freundlich gesinnt, wenn dieser das Ei gestohlen hat und auf dem Weg zur Hütte ist. Dann kann es nämlich sein, daß der Feuerstrahl des Sauriers unseren Leo am Leben läßt.

Wenn Leo das Ei gestohlen hat, so kann er nur noch Richtung Hütte gehen und nicht mehr in die Richtung des Sauriers.

Schwierigkeitssteigerung:

Die anfänglich gewählte Schwierigkeitsstufe steigert sich nach jeweils 12 gesammelten Eiern.

Wenn es Ihnen gelingt die gesamte Höhle mit Eiern zu füllen (24 Eier), so bekommen Sie einen Bonushammer (maximal können 6 Hämmer angezeigt werden).

Punktwertung:

Sie erhalten für jedes gesammelte Ei 10 mal Schwierigkeitsstufe Punkte. Gelingt es Ihnen die gesamte Höhle mit Eiern zu füllen, so werden Ihnen neben dem Bonushammer zusätzlich 1000 mal der augenblicklichen Schwierigkeit Punkte gutgeschrieben.

Nachdem das Spiel zu Ende ist, können Sie durch Druck auf den linken Aktionsknopf ein neues Spiel beginnen.

Tips zum Eintippen:

Verwenden Sie soviel als möglich die (CTRL) Tasten, um Tippfehler auszuschließen.

An Stelle von „Print" kann einfach ? eingegeben werden.

Sämtliche Leerzeichen in den Basic-Anweisungen können weggelassen werden, da diese vom Computer automatisch generiert werden.

Saven Sie das Programm nach dem Eintippen mehrmals auf Cassette oder Diskette weg, damit Sie im Falle eines Systemabsturzes nicht das ganze Programm nochmals eintippen müssen.

```
10  REM ***************************
12  REM *   C A V E    M A N     *
14  REM *-----------------------*
15  REM * LASER 2001+li.Joystick *
16  REM *-----------------------*
18  REM * (C) 1984 by Tronicsoft *
20  REM *            V.Becker    *
22  REM ***************************
24  REM
26  REM
100  TEXT
110  COLOR= 13,1
120  HOME
130  PRINT  SPC( 10);"C A V E    M A N"
140  PRINT  SPC( 10);"---------------"
150  PRINT  CHR$ (24); CHR$ (24); CHR$ (24)
```

```
160  PRINT "Bitte waehlen Sie die Schwierigkeit"; CHR$ (24); CHR$ (24)
170  PRINT  SPC( 4);"1 - kleinste Schwierigkeit"
180  PRINT  SPC( 4);"8 - groesste Schwierigkeit"
190  FOR I = 1 TO 9: PRINT  CHR$ (24);: NEXT I
200  PRINT  SPC( 7);"(C) 1984 by Tronicsoft"
205  GOSUB 60000
210  GET RU
220  IF RU < 1 OR RU > 8 THEN 210
1000  GOSUB 50000
1005 VH = 0:SE = 0:FI = 0:H = 3:P = 0:E = 0
1010 PH = 5:PM = 7:AE = 0:E = 0
1020 SH = PH * 8:SP = PM *.8 + 1
1030 ZH = 150:ZM = 139
1035 S = 202:Z = 152: GOSUB 55000
1036  GOSUB 10450
1037  GOSUB 40000
1040  GOSUB 56000
1041  RESTORE
1042  GOSUB 60000:EN = 7: GOSUB 20000: GOSUB 60000
1043  GOSUB 25000
1045  FOR LI = 1 TO 9 - RU
1050  CALL  - 1536
1060 JS =  PEEK (24)
1065  IF VH = 2 THEN 10800
1070  IF JS = 0 THEN 2000
1080  IF JS > 100 THEN 10600
1090  GOTO 10000
2000  IF FI = 1 AND PM > 22 THEN 35000
2010  NEXT LI
2015  IF AS > 0 THEN AS = AS - 1: GOSUB 59100: GOTO 2040
2017  GOSUB 59400
2020 SV = SV + 1: IF SV > 4 THEN SV = 1
2030  ON SV GOSUB 51000,52000,53000,54000
2040  GOTO 1045
10000 JS = JS / 2
10010  IF JS > 10 THEN 11000
10015  FOR K = 1 TO 7
10020  ON JS GOTO 2000,10100,10100,2000,10100
10100  IF SE = 1 THEN 2000
10102  IF VH = 1 THEN 10150
10105  IF PM > 23 THEN 10120
10110  GOSUB 57500:PM = PM + 1:SP = PM * 8 + 1: GOSUB 56000
10115  GOTO 2000
10120  GOSUB 55500:PM = 23: GOSUB 57500: GOSUB 57000
10125  ON SV GOSUB 51000,52000,53000,54000
10130 S = (PM - 1) * 8:Z = 152: GOSUB 55000:SE = 1
10140  GOTO 2000
10150  IF PM > 22 THEN  GOSUB 59000:VH = 0: GOSUB 49900: GOTO 2000
10160  GOSUB 59000: GOSUB 57500
10170 PM = PM + 1:SP = PM * 8 + 1: GOSUB 56000
10180 SH = (PM + 1) * 8:ZH = 141: GOSUB 58000
10190  GOTO 2000
10200  IF PM < 6 THEN 2000
10205  IF VH = 1 THEN 2000
10210  IF SE = 1 THEN  GOSUB 55500
10220  GOSUB 57500:PM = PM - 1:SP = PM * 8 + 1: GOSUB 57000
10225  IF PM < 22 THEN 10230
10226  ON SV GOSUB 51000,52000,53000,54000
10230  IF SE = 1 THEN 10250
10240  GOTO 2000
10250 S = (PM - 2) * 8:Z = 152: GOSUB 55000
10260  IF PM = 6 THEN 10300
10270  GOTO 2000
10300  GOSUB 55500
10310  FOR I = 20 TO 21:Z = 8 * I + 3
10320  GOSUB 55000
10330  SOUND (195,1,15)
10340  GOSUB 55500
10350  NEXT I
10360  FOR I = 4 TO 27 - AE:S = 8 * I
10370  GOSUB 55000
10375  SOUND (137,1,15)
```

```
10380  GOSUB 55500
10390  NEXT I
10400  GOSUB 55000
10405 P = P + 10 * RU: GOSUB 40000
10410 AE = AE + 1:SE = 0:E = E + 1
10415 S = 202:Z = 152: GOSUB 55000
10417  IF E = 12 THEN  GOSUB 10430
10420  GOTO 2000
10430 RU = RU + 1: IF RU > 8 THEN RU = 8
10432  GOSUB 60100
10435 E = 0
10440  IF AE = 24 THEN H = H + 1
10450  IF H > 6 THEN H = 6
10460  IF H > 0 THEN 10480
10470  GOTO 10496
10480  FOR K = 8 TO 8 + H - 1
10490 SH = 8 * K:ZH = 20: GOSUB 58000
10495  NEXT K
10496  IF AE = 24 THEN AE = 0: GOSUB 10510
10500  RETURN
10510  COLOR= 15,1
10512  GOSUB 60000
10514 P = P + 1000 * RU: GOSUB 40000
10520  FOR I = 171 TO 176: UNPLOT 32,I TO 223,I
10530  NEXT I
10540  RETURN
10600  IF H = 0 THEN 2000
10610  IF PM < 9 AND H > 0 THEN 10630
10620  GOTO 10700
10630  IF VH > 0 THEN 2000
10635 VH = 1: GOSUB 57500: GOSUB 56000
10640 SH = (PM + 1) * 8:ZH = 141: GOSUB 58000
10650  GOTO 2000
10700  IF VH = 0 THEN 2000
10705  GOSUB 59000
10710 VH = 2:ZH = 129:X =  - 4: GOSUB 58000
10720  GOTO 2000
10800  IF VH = 0 THEN 2000
10805  GOSUB 59000
10810 ZH = ZH + X:SH = SH + RU: GOSUB 58000
10820  IF ZH = 105 THEN X =  - X
10830  IF SH > 230 THEN 10900
10840  IF SH > 192 AND ZH > 125 THEN 10950
10850  IF ZH = 129 THEN 10900
10860  GOTO 1070
10900 VH = 0: GOSUB 59000: GOSUB 49900: GOTO 2000
10950  GOSUB 59000:VH = 0:AS = 12 - RU: GOSUB 54000: GOSUB 51000: GOSUB 52000:SV
 = 2: GOTO 2000
11000 JS = JS - 15
11010  ON JS GOTO 10200,10200,2000,2000,10200
20000  DATA 64,1,80,2,96,3,112,4,144,5,160,2,176,6
20005  DATA 56,7,72,2,88,5,104,4,136,8,152,3,168,4,184,9
20010 C = 0
20015  FOR K = 1 TO EN
20016 C = C + 2: IF C > 15 THEN C = 3
20020  COLOR= C,1: READ PV,SK
20030  FOR I = 0 TO 2
20040  ON SK GOSUB 20100,20200,20300,20400,20500,20600,20700,20800,20900
20050  NEXT I
20060  FOR U1 = PV TO PV + 14: FOR U2 = 1 TO 3
20070 U =  RND (255) * 15: UNPLOT U1,105 + U
20080  NEXT U2: NEXT U1
20090  NEXT K
20095  RETURN
20100  PLOT PV + 12 + I,107 - I TO PV + I,107 - I TO PV + I,119.- I TO PV + 12 +
 I,119 - I
20110  RETURN
20200  PLOT PV + I,119 - I TO PV + 6 + I,107 - I TO PV + 12 + I,119 - I
20210  PLOT PV + 3 + I,116 - I TO PV + 9 + I,116 - I
20220  RETURN
20300  PLOT PV + I,107 - I TO PV + 6 + I,119 - I TO PV + 12 + I,107 - I
20310  RETURN
```

```
20400  PLOT PV + 12 + I,107 - I TO PV + I,107 - I TO PV + I,119 - I TO PV + 12 +
 I,119 - I
20405  PLOT PV + I,113 - I TO PV + 6 + I,113 - I
20410  RETURN
20500  PLOT PV + I,119 - I TO PV + I,107 - I TO PV + 6 + I,115 - I TO PV + 12 +
I,107 - I TO PV + 12 + I,119 - I
20510  RETURN
20600  PLOT PV + I,119 - I TO PV + I,107 - I TO PV + 12 + I,119 - I TO PV + 12 +
 I,107 - I
20610  RETURN
20700  PLOT PV + 12 + I,107 - I TO PV + I,107 - I TO PV + I,119 - I TO PV + 12 +
 I,119 - I TO PV + 12 + I,113 - I TO PV + 3 + I,113 - I
20710  RETURN
20800  RECT (PV + I,107 - I),(PV + 12 + I,119 - I)
20810  RETURN
20900  PLOT PV + I,119 - I TO PV + I,107 - I TO PV + 9 + I,107 - I TO PV + 12 +
I,109 - I TO PV + 12 + I,111 - I TO PV + 9 + I,113 - I TO PV + 3 + I,113 - I TO
PV + 12 + I,119 - I
20910  RETURN
25000  COLOR= 1,1
25005  CALL  - 1536: IF  PEEK (24) > 100 THEN 25020
25010  GOTO 25005
25020  FOR I = 104 TO 120: UNPLOT 64,I TO 191,I
25030  NEXT I
25040  RETURN
30000  RECT (PV + 1,21),(PV + 6,27): RETURN
30400  PLOT PV + 1,21 TO PV + 1,24 TO PV + 6,24 TO PV + 6,21
30410  PLOT PV + 6,25 TO PV + 6,27
30420  RETURN
30500  PLOT PV + 6,21 TO PV + 1,21 TO PV + 1,24 TO PV + 6,24 TO PV + 6,27 TO PV
+ 1,27
30510  RETURN
30600  RECT (PV + 1,24),(PV + 6,27): PLOT PV + 1,23 TO PV + 1,21 TO PV + 1,21
30610  RETURN
30700  PLOT PV + 1,21 TO PV + 6,21 TO PV + 6,27
30710  RETURN
30800  RECT (PV + 1,21),(PV + 6,27): PLOT PV + 2,24 TO PV + 5,24
30810  RETURN
30900  RECT (PV + 1,21),(PV + 6,24): PLOT PV + 6,25 TO PV + 6,27 TO PV + 1,27
30910  RETURN
31000  PLOT PV + 6,21 TO PV + 6,27
31010  RETURN
32000  PLOT PV + 1,21 TO PV + 6,21 TO PV + 6,24 TO PV + 1,24 TO PV + 1,27 TO PV
+ 6,27
32010  RETURN
33000  PLOT PV + 1,21 TO PV + 6,21 TO PV + 6,27 TO PV + 1,27
33010  PLOT PV + 1,24 TO PV + 5,24
33020  RETURN
35000  RESTORE
35005  GOSUB 60000
35010  FOR I = 1 TO 7: READ A,B: NEXT I
35020 EN = 8: GOSUB 20000
35030  GOSUB 60000
35040  CALL  - 1536: IF  PEEK (24) < 100 THEN 35040
35050  RUN
40000  COLOR= 8,1
40005  IF P >  = 100000 THEN P = 99999
40010 P$ =  STR$ (P)
40020 PV = 192 - 8 *  LEN (P$)
40030 L =  VAL ( MID$ (P$,1,1)) + 1
40050  UNPLOT PV + 1,21 TO PV + 6,21 TO PV + 6,27 TO PV + 1,27 TO PV + 1,21
40060  UNPLOT PV + 2,24 TO PV + 5,24
40070  ON L GOSUB 30000,31000,32000,33000,30400,30500,30600,30700,30800,30900
40080  IF  LEN (P$) <  = 1 THEN  RETURN
40090 P$ =  RIGHT$ (P$, LEN (P$) - 1)
40100  GOTO 40020
49900 H = H - 1:SH = (8 + H) * 8:ZH = 20: GOSUB 59000
49910  IF H = 0 THEN 35000
49920  RETURN
50000  GR
50010 C = 2
50020  FOR K = 0 TO 8: FOR I = 0 TO 250 STEP 8
```

```
50030  COLOR= C,1:C = C + 2: IF C > 15 THEN C = 2
50040  PLOT I,K TO I + 7,K
50060  PLOT I,191 - K TO I + 7,191 - K
50070  NEXT I: NEXT K
50080  FOR K = 9 TO 183: FOR I = 0 TO 1
50090  COLOR= C,1:C = C + 2: IF C > 15 THEN C = C - 15 + 1
50100  PLOT I * 8,K TO I * 8 + 7,K
50120  PLOT 240 + I * 8,K TO 240 + I * 8 + 7,K
50130  NEXT I: NEXT K
50140  COLOR= 13,1
50150  FOR I = 16 TO 239: PLOT I,182 TO I,182 -  RND (255) * 4
50160  NEXT I
50170  FOR I = 16 TO 239: PLOT I,166 TO I,166 +  RND (255) * 4
50180  NEXT I
50182  FOR I = 161 TO 165: PLOT 16,I TO 239,I
50184  NEXT I
50186  COLOR= 10,1
50190  FOR I = 16 TO 239: PLOT I,160 TO I,160 -  RND (255) * 2
50200  NEXT I
50205  COLOR= 13,1
50210  FOR I = 29 TO 39: UNPLOT I,157 TO I,170
50220  NEXT I
50230  FOR I = 166 TO 182: PLOT 16,I TO 28,I
50240  PLOT 228,I TO 239,I
50250  NEXT I
50260  FOR I = 160 TO 182: PLOT 28,I TO 28 +  RND (255) * 4,I
50270  NEXT I
50280  FOR I = 166 TO 182: PLOT 228,I TO 228 -  RND (255) * 4,I
50290  NEXT I
50320  COLOR= 8,1
50330  FOR I = 24 TO 28: PLOT I,160 TO I,128
50340  NEXT I
50350  FOR I = 1 TO 2: CIRCLE (26,127),I: NEXT I
50360  FOR I = 24 TO 28: FOR K = 1 TO 10:U =  RND (255) * 30:U = 128 + U: UNPLOT
 I,U
50370  NEXT K: NEXT I
50380  FOR I = 129 TO 133: PLOT 23,I TO 63,I - 4
50390  NEXT I
50400  FOR I = 1 TO 2: CIRCLE (22,132),I: CIRCLE (64,127),I: NEXT I
50410  FOR I = 126 TO 133: FOR K = 1 TO 12:U =  RND (255) * 40:U = 23 + U: UNPLO
T U,I
50420  NEXT K: NEXT I
50430  COLOR= 4,1
50440  FOR I = 1 TO 40
50450  PLOT 16,80 + I TO 24,70 + I TO 32,75 + I TO 40,60 + I TO 48,58 + I TO 56,
52 + I TO 64,45 + I TO 72,39 + I
50460  PLOT 72,39 + I TO 80,47 + I TO 88,49 + I TO 96,54 + I TO 104,68 + I TO 11
2,50 + I TO 120,39 + I TO 128,42 + I
50470  PLOT 128,42 + I TO 136,51 + I TO 144,64 + I TO 152,66 + I TO 160,60 + I T
O 168,56 + I TO 176,50 + I TO 184,53 + I
50480  PLOT 184,53 + I TO 192,54 + I TO 200,59 + I TO 208,63 + I TO 216,60 + I T
O 224,70 + I TO 232,62 + I TO 239,80 + I
50500  NEXT I
50510  FOR I = 80 TO 90: PLOT 16,I TO 239,I
50520  NEXT I
50530  FOR I = 91 TO 120: UNPLOT 16,I TO 239,I
50540  NEXT I
50550  FOR I = 16 TO 239: PLOT I,91 TO I,91 +  RND (255) * 3
50560  NEXT I
50600  UNPLOT 32,76 TO 40,80 TO 48,92
50610  UNPLOT 104,68 TO 96,75 TO 88,88
50620  UNPLOT 152,67 TO 160,80 TO 168,90
50630  UNPLOT 224,70 TO 216,73 TO 208,80 TO 200,83
50640  COLOR= 14,1
50650  FOR I = 6 TO 26 STEP 10
50655 J = 8 * I
50660  FOR K = 1 TO 5
50670  CIRCLE (J,22),K: CIRCLE (J - 4,24),K: CIRCLE (J + 4,23),K
50680  NEXT K: NEXT I
50690  COLOR= 2,1
50700  FOR I = 0 TO 2
50710  PLOT 215 + I,139 TO 215 + I,141 TO 210 + I,146 TO 210 + I,148 TO 209 + I,
```

```
149 TO 209 + I,152 TO 213 + I,156 TO 222 + I,156
50720  PLOT 227 - I,156 TO 230 - I,153 TO 233 - I,153 TO 238 - I,145 TO 232 - I,
149 TO 228 - I,149 TO 229 - I,146 TO 225 - I,147
50730  PLOT 225 - I,147 TO 222 - I,144 TO 222 - I,141 TO 220 - I,141 TO 218 - I,
139
50740  NEXT I
50750  FOR I = 212 TO 222: PLOT I,145 TO I,155
50760  NEXT I
50770  FOR I = 142 TO 144: PLOT 215,I TO 222,I
50780  NEXT I
50790  FOR I = 223 TO 227: PLOT I,147 TO I,155
50800  NEXT I
50810  FOR I = 150 TO 153: PLOT 228,I TO 234,I
50820  NEXT I
50840  FOR I = 157 TO 159: PLOT 215,I TO 218,I
50850  PLOT 223,I TO 226,I
50860  NEXT I
50870  GOSUB 51000
50900  RETURN
51000  COLOR= 2,1
51006  FOR I = 135 TO 143: UNPLOT 203,I TO 211,I
51007  NEXT I
51008  UNPLOT 217,128 TO 217,130 TO 218,130 TO 218,136 TO 219,137 TO 219,130
51010  PLOT 211,127 TO 215,127 TO 216,128 TO 210,128 TO 209,129 TO 210,129
51020  PLOT 213,129 TO 216,129 TO 216,130 TO 204,130 TO 203,131 TO 217,131 TO 21
7,132 TO 208,132
51030  PLOT 217,133 TO 204,133 TO 205,134 TO 217,134 TO 217,135 TO 211,135
51040  FOR I = 0 TO 2: PLOT 213 + I,136 + I TO 217 + I,136 + I
51050  NEXT I
51090  RETURN
52000  COLOR= 2,1
52010  PLOT 207,136 TO 208,135 TO 209,135 TO 203,141 TO 203,142 TO 210,135 TO 21
0,136 TO 209,137
52120  PLOT 211,136 TO 204,143 TO 205,143 TO 212,136
52130  PLOT 217,128 TO 217,130 TO 218,130 TO 218,136 TO 219,137 TO 219,130
52140  UNPLOT 211,127 TO 215,127 TO 215,128 TO 210,128 TO 209,129 TO 213,129
52150  UNPLOT 211,130 TO 204,130 TO 203,131 TO 209,131
52160  UNPLOT 203,133 TO 207,133 TO 207,134 TO 204,134
52170  UNPLOT 210,133 TO 209,134
52180  UNPLOT 214,136 TO 214,137
52190  RETURN
53000  COLOR= 9,1
53005 FI = 1
53010  FOR I = 186 TO 192 STEP 2: PLOT 202,144 TO I,157
53020  NEXT I
53030  SOUND (1,1,15),(2,1,15),(3,1,15),(1,1,15,1)
53040  RETURN
54000  COLOR= 9,1
54005 FI = 0
54010  FOR I = 186 TO 192 STEP 2: UNPLOT 202,144 TO I,157
54020  NEXT I
54040  RETURN
55000  COLOR= 15,1
55010  PLOT S + 2,Z TO S + 4,Z + 1 TO S + 2,Z + 1 TO S,Z + 2 TO S + 5,Z + 2 TO S
 + 5,Z + 3 TO S,Z + 3 TO S,Z + 4 TO S + 5,Z + 4 TO S + 4,Z + 5 TO S + 1,Z + 5
55015  SOUND (113,1,15)
55020  RETURN
55500  COLOR= 15,1
55510  UNPLOT S + 2,Z TO S + 4,Z + 1 TO S + 2,Z + 1 TO S,Z + 2 TO S + 5,Z + 2 TO
 S + 5,Z + 3 TO S,Z + 3 TO S,Z + 4 TO S + 5,Z + 4 TO S + 4,Z + 5 TO S + 1,Z + 5
55515  SOUND (16,1,15)
55520  RETURN
56000  COLOR= 10,1
56005 SM = SP - 1
56010  PLOT SM,ZM TO SM + 2,ZM
56020  PLOT SM - 1,ZM + 1 TO SM + 3,ZM + 1
56030  FOR I =  - 2 TO  - 1: PLOT SM + I,ZM + 2 TO SM + I,ZM + 4
56040  NEXT I
56050  COLOR= 6,1
56060  FOR I = 2 TO 4: PLOT SM,ZM + I TO SM + 3,ZM + I
56070  NEXT I
56080  UNPLOT SM + 2,ZM + 2
```

```
56090  PLOT SM + 4,ZM + 3
56100  PLOT SM,ZM + 5 TO SM + 2,ZM + 5
56120  PLOT SM,ZM + 6 TO SM + 1,ZM + 6
56130  PLOT SM,ZM + 7 TO SM + 2,ZM + 7
56140  FOR I = 8 TO 11: PLOT SM - 1,ZM + I TO SM + 3,ZM + I
56150  NEXT I
56160  PLOT SM + 4,ZM + 8 TO SM + 4,ZM + 7 TO SM + 7,ZM + 7
56170  PLOT SM + 6,ZM + 6 TO SM + 6,ZM + 8
56180  COLOR= 7,1
56190  PLOT SM - 1,ZM + 12 TO SM + 3,ZM + 12 TO SM + 3,ZM + 13 TO SM - 1,ZM + 13
 TO SM - 1,ZM + 14 TO SM,ZM + 14
56200  PLOT SM + 2,ZM + 14 TO SM + 3,ZM + 14
56210  COLOR= 6,1
56220  FOR I = 15 TO 16: PLOT SM - 1,ZM + I TO SM,ZM + I
56230  PLOT SM + 2,ZM + I TO SM + 3,ZM + I
56240  NEXT I
56250  FOR I = 17 TO 18: PLOT SM - 1,ZM + I TO SM + 4,ZM + I
56260  NEXT I
56265  SOUND (135,1,15)
56270  RETURN
57000  COLOR= 10,1
57005 SL = SP - 2
57010  PLOT SL,ZM TO SL - 2,ZM
57020  PLOT SL + 1,ZM + 1 TO SL - 3,ZM + 1
57030  FOR I =  - 2 TO  - 1: PLOT SL - I,ZM + 2 TO SL - I,ZM + 4
57040  NEXT I
57050  COLOR= 6,1
57060  FOR I = 2 TO 4: PLOT SL,ZM + I TO SL - 3,ZM + I
57070  NEXT I
57080  UNPLOT SL - 2,ZM + 2
57090  PLOT SL - 4,ZM + 3
57100  PLOT SL,ZM + 5 TO SL - 2,ZM + 5
57120  PLOT SL,ZM + 6 TO SL - 1,ZM + 6
57130  PLOT SL,ZM + 7 TO SL - 2,ZM + 7
57140  FOR I = 8 TO 11: PLOT SL + 1,ZM + I TO SL - 3,ZM + I
57150  NEXT I
57160  PLOT SL - 4,ZM + 8 TO SL - 4,ZM + 7 TO SL - 7,ZM + 7
57170  PLOT SL - 6,ZM + 6 TO SL - 6,ZM + 8
57180  COLOR= 7,1
57190  PLOT SL + 1,ZM + 12 TO SL - 3,ZM + 12 TO SL - 3,ZM + 13 TO SL + 1,ZM + 13
 TO SL + 1,ZM + 14 TO SL,ZM + 14
57200  PLOT SL - 2,ZM + 14 TO SL - 3,ZM + 14
57210  COLOR= 6,1
57220  FOR I = 15 TO 16: PLOT SL + 1,ZM + I TO SL,ZM + I
57230  PLOT SL - 2,ZM + I TO SL - 3,ZM + I
57240  NEXT I
57250  FOR I = 17 TO 18: PLOT SL + 1,ZM + I TO SL - 4,ZM + I
57260  NEXT I
57265  SOUND (135,1,15)
57270  RETURN
57500  COLOR= 1,1
57505 SM = SP
57510  FOR I =  - 8 TO 7: UNPLOT SM + I,ZM TO SM + I,ZM + 18
57520  NEXT I
57530  SOUND (73,1,15)
57540  RETURN
58000  COLOR= 8,1
58010  PLOT SH + 3,ZH TO SH + 3,ZH + 7 TO SH + 4,ZH + 7 TO SH + 4,ZH
58020  COLOR= 14,1
58030  FOR I = 1 TO 3: PLOT SH + I,ZH + I TO SH + 6,ZH + I
58040  NEXT I
58045  SOUND (184,1,15)
58050  RETURN
59000  COLOR= 8,1
59010  UNPLOT SH + 3,ZH TO SH + 3,ZH + 7 TO SH + 4,ZH + 7 TO SH + 4,ZH
59020  COLOR= 14,1
59030  FOR I = 1 TO 3: UNPLOT SH + I,ZH + I TO SH + 6,ZH + I
59040  NEXT I
59045  SOUND (165,1,15)
59050  RETURN
59100 SS = SS + 1: IF SS > 2 THEN SS = 1
```

```
59115 FI = 0
59120  ON SS GOSUB 59200,59300
59130  RETURN
59200  GOSUB 59400
59220  PLOT 211,126 TO 213,124
59230  PLOT 213,126 TO 211,124
59240  PLOT 216,126 TO 218,124
59250  PLOT 218,126 TO 216,124
59255  SOUND (211,1,15)
59260  RETURN
59300  GOSUB 59400
59310  COLOR= 8,1
59340  PLOT 214,126 TO 216,124
59345  PLOT 216,124 TO 214,124
59347  SOUND (195,1,15)
59360  RETURN
59400  COLOR= 8,1
59410  FOR L = 124 TO 126: UNPLOT 211,L TO 218,L
59420  NEXT L
59430  RETURN
60000  SOUND (135,5,15),(160,5,15)
60010  SOUND (95,5,15)
60020  SOUND (113,5,15)
60030  SOUND (135,10,15)
60040  SOUND (148,10,15)
60050  SOUND (135,10,15)
60060  SOUND (65,5,15),(95,5,15)
60070  SOUND (95,5,15),(135,5,15)
60080  RETURN
60100  SOUND (95,8,15),(135,8,15),(160,8,15)
60110  SOUND (113,8,15),(135,8,15),(165,8,15)
60120  SOUND (95,8,15),(135,8,15),(160,8,15)
60130  RETURN
```

Farben
3 Tonkanäle und 1 Rauschkanal
32 Sprite möglich
eingebautes Joystickinterface
eingebaute parallele Schnittstelle
Kassetteninterface

Diskettenstation
2 Laufwerke steuerbar
Standard 5,25 inch einseitige Diskette
Spurendichte: 48 tpi
Spuren pro Diskettenseite: 40
Sektoren pro Spur: 16
Bytes pro Sektor: 256
Kapazität: 160 KB pro Diskettenseite
Plattenrotationsgeschwindigkeit: 300 rpm

Unverbindliche Preisempfehlungen:

LASER 2001 Konsole:	DM 598,-
16 K Memory	
Expansion Modul:	DM 218,-
DR 10 Datenrecorder:	DM 99,-
PC 20 Printer Cable:	DM 69,-
DI 100 A Disc Drive Controller +	
FD 100 A Floppy Dics Drive:	DM 698,-

Wir danken der Fa. Sanyo Video für die freundliche Bereitstellung der Testgeräte und für die freundliche Unterstützung bei technischen Fragen.

Volker Becker

Preiswert drucken, aber?

Um mit dem TI-99-4A im DIN-A 4-Format zu drucken, war bisher eine aufwendige, kostenintensive Peripherie erforderlich. Nun aber geht es einfacher und preiswerter.

Es gibt verschiedene Interfaces, die direkt an die Konsole angeschlossen werden können, und mit diesen kann dann ein Drucker direkt verbunden werden. Für diesen Bericht wurden getestet:
Das Interface MBI der Fa. Klein.
Der Drucker stx-80 der Fa. star.
Die Montage beider Geräte dauert keine 5 Minuten.
Bevor wir die Arbeitsweise beschreiben, sollen zwei „Nachteile" des Druckkers nicht verschwiegen werden:

1. Es wird Thermopapier benutzt.
2. Die Bedienungsanleitung ist dürftig und nur in englisch.

Zu 1 muß man aber bedenken, daß Farbbänder für Normaldruck auch nicht eben billig sind und Reparaturen am Thermodrucker wegen der nichtvorhandenen komplizierten Mechanik, z. B. des Druckkopfes, seltener und preiswerter sein dürften.
Zu 2. Soll dieser Bericht Abhilfe schaffen. Wir wollen nicht zuviel technische Daten bringen, sondern praktische Hilfe geben, die aus der Anleitung nicht oder nur schwer zu entnehmen ist.
Was leistet der stx-80 nun? – Da wären zu erwähnen:
Sehr leiser Druck mit 60 Zeichen/sec.
80 Zeichen pro Zeile, in Breitschrift 40.
Deutsche Umlaute Ä, Ö, Ü und ß.
Groß- und Kleinschrift (siehe Bild 1).
Breitdruck (siehe Bild 2).
64 vordefinierte Grafikzeichen (siehe Bild 3 u. 4).
Bit Image Mode (Einzelpunktansteuerung).

Änderung des Zeilenabstandes (wichtig z. B. bei Balkendiagrammen)

Alle diese Eigenschaften können nach Einstellung der DIP-Schalter softwaremäßig gesteuert werden. Genau damit wollen wir uns befassen. Denn hier liegt die Schwäche des Manuals.
Das beste Schriftbild ergibt sich, wenn man in Großbuchstaben schreibt. Daher braucht man auch nur über die eckige Klammer (auf) ein Ä, über dem Schrägstrich links unten ein Ö (Z-Taste) und über der eckigen Klammer (zu) ein Ü zu kleben.
Angesprochen wird der Drucker mit „RS232.BA=4800" ohne Zwischenraum! Die 1. PGM.-Zeile heißt dann:
100 open #1:„RS232.BA=4800", output,
und gedruckt wird mit:
110 Print#1: „Testtext".
Will man dieses Wort in Breitschrift drucken, so heißt der Befehl: CHR$(14); also:
110 Print#1:CHR$(14); „Testext".
Der Befehl CHR$(14) muß vor jeder Zeile stehen, die dann die Hälfte nämlich statt 80 Zeichen hat. Soll hinter der Breitschrift wieder in Normalschrift gedruckt werden, muß der Befehl mit CHR '
CHR#(20); wieder aufgehoben werden. Also:
110 Print#1:CHR$(14);„Testext"; CHR$(20); „Testext".
Zu beachten ist das Zusammentreffen von Grafik und Breitschrift. Hier müssen die Befehle in aufsteigender Reihenfolge eingegeben werden. Grafik=(15):
110 Print#1:CHR$(14); CHR$(15); „Testext".

Und hier nun die Liste der Steuerbefehle:

CHR$(9); = Druck ab Spalte 11
CHR$(10); = 2 Leerzeilen
CHR$(12); = Vorschub auf die nächste Seite
CHR$(13); = 1 Leerzeile
CHR$(14); = Breitschriftmodus
CHR$(15); = Grafikmodus
CHR$(18); = Aufhebung des Grafikmodus
CHR$(20); = Aufhebung des Breitschriftmodus
CHR$(27);„A"CHR$(1); = Zeilenabstand 1/11 Zoll
CHR$(27);„A"CHR$(2); = Zeilenabstand 1/6 Zoll

1/11 Zoll bedeutet praktisch keinen Abstand zwischen den Zeilen und wird z. B. bei den Grafiken benutzt. 1/6 Zoll ist die Rückstellung in den Normalabstand.
Achten Sie darauf, den Grafikmodus immer aufzuheben, da der Drucker sonst alles folgende ebenfalls im Grafikmodus ausgibt (nicht zu verwechseln mit dem Bit Image Modus).
Eine Tabelle der Grafikzeichen und welches Tastaturzeichen sie anschlagen müssen, zeigen die Bilder 3 und 4. Bild 4 zeigt die Wiedergabe durch den Drucker. Bei Bild 3 helfen die Umrandungen, die Zeichen, z. B. senkrechte Striche, richtig zu plazieren.
Um beispielsweise Balkendiagramme auszudrucken, darf praktisch kein Abstand zwischen den Zeilen bestehen. Dazu dient der vorletzte Befehl aus der vorstehenden Liste. Wird hierbei mit For-Next-Schleifen gearbeitet, muß der Befehl
CHRY$(27);„A";CHR$(1);
vor der Schleife stehen und der ihn aufhebende Befehl dahinter.
Die Stellung der DIP-Schalter, die leicht zugänglich unter Papierrolle liegen, ist für die deutsche Schreibweise:
1, 3 und 5 nach hinten (ON),
2, 4 und 6 nach vorne (OFF.
Damit kennen Sie die wichtigsten Eigenschaften und können nun Ihren TI noch vielseitiger einsetzen.
Zum Schluß noch ein Lob der Fa. star, die dem Autor, der sich als Privatmann meldete, bei einigen der obigen Probleme mit telefonischen Rückrufen bei der Lösung geholfen hat. Da schöpft man, weil ungewöhnlich in der Branche, gleich wieder etwas Hoffnung.

Herbert May

star Europe GmbH
Frankfurter Allee 1–3
6236 Eschborn-TS
Klein-Computer
Münzstraße 34 A
5600 Wuppertal 2

Neue Produktlinie Zubehör für alle PC-Computer-Systeme unter der Marke ASCOM

Zur Hifi '84 hat die DYNAMICS marketing GmbH ihr Angebot ganz wesentlich erweitert. Unter der Marke ASCOM bringt DYNAMICS ein umfangreiches Computer-Zubehör-Programm auch für den PC-Bereich auf den Markt.
Die einzelnen Produktgruppen sind

1. Kabel
2. Buchsen, Stecker, Kupplungen
3. Interface, Buffer, Weiche
4. Akustik-Koppler, Lightpen
5. Computer-Pflege-Serie

1. Kabel

Zum Verlängern, Anschließen und Verbinden des Computers mit Peripheriegeräten sind gute und passende Kabel unerläßlich, deshalb bietet Dynamics unter der Marke ASCOM folgende Kabel an:
Computer-Rundkabel, 9adrig und Computer-Rundkabel 25adrig, beide mit Abschirmung gegen Störungen z. B. von benachbarten Geräten, 2 m Länge.
Computer-Flachband-Kabel, 25adrig, flache Kabel sparen Platz, sind mechanisch flexibler und widerstehen besser äußerem Druck.

2. Buchsen, Stecker, Kupplungen
Für die Verbindung zwischen diversen PC-Computern mit den üblichen Schnittstellen und Peripheriegeräten wie Drucker, Akustikkoppler usw. braucht man Buchsen, Stecker und Kupplungen. Alle Kontakte sind Gold/Zinn plattiert. Es gibt im einzelnen:
RS 232C/V24 Rundkabel-Stecker, 25polig mit Steckerkappe,
RS 232C/V24 Rundkabel-Stecker-Buchse, 25polig mit Buchsenkappe;
Centronics-Rundkabel-Stecker, 36polig mit Abschirmung und Metallkappe,
Centronics-Rundkabel-Stecker, 36polig mit Abschirmung und Metallkappe;
RS 232C/V24 Wechselkupplung, 25polig, Stecker/Stecker, durchkontaktiert,
RS 232C/V24 Wechselkupplung, 25polig, Buchse/Buchse, durchkontaktiert.

3. Interface, Buffer, Weiche
RS 232C/V24-Interface für den Commodore 64 zum Anschließen von Peripherie-Geräten, mit durchgeführtem User-Port, außerdem schaltbar: RS 232C/V24 User-Port.
2K-Spooler für den Commodore 64, gepuffertes Drucker-Interface für alle RS 232C/V24 Schnittstellen, mit durchgeführtem User-Port, erweiterbar mit 16K-Modules bis auf 34K.
16K-Erweiterungsbuffer für Spooler.
IEEE-488-Interface für Commodore 64, 8032 und 4032 zum Anschluß von IEEE-Geräten, mit durchgeführtem Extentions-Bus.
RS 232C/V24 Weiche für den Anschluß an die RS 232C/V24 Schnittstelle zum Koppeln von 2 Peripherie-Geräten, umschaltbar.
Centronics-Weiche für den Anschluß an Centronics-Schnittstellen zum Koppeln von 2 Peripherie-Geräten, umschaltbar.

4. Akustik-Koppler und Lightpen
Mit den bewährten Competition Pro Joysticks und dem Disketten-Locher hat sich Dynamics beim Verbraucher bereits einen Namen als Anbieter für interessantes Zubehör geschaffen. Dies Verbraucherinteresse soll mit den neuen Produkten gefestigt und ausgebaut werden.
Zum Herbst- und Weihnachtsgeschäft präsentiert Dynamics den Akustik-Koppler von Ascom für den Commodore 64 mit integriertem Interface zum direkten Anschluß auf dem Extention-Bus des C64. Der Ext.-Bus wird im Akustik-Koppler durchgeführt, so daß der Anwender weiterhin den Ext.-Bus für andere Einsatzzwecke benutzen kann.
Eine FTZ-Nr. wurde bereits beantragt. Besonders interessant aber ist der absolut günstige empf. VK-Preis von 198,- DM.

5. Computer-Pflege-Serie
Um Störungen der gegen Verunreinigungen sehr empfindlichen Computer-Hardware zu vermeiden, ist die richtige Pflege und der Schutz vor Staub für den Computer-Besitzer besonders wichtig.
Die Ascom-52/2"-Reinigungsdiskette, mit der Reinigungslösung in einer separaten Flasche, reinigt Lese- und Schreibköpfe optimal, sie gehört zur regelmäßigen Wartung der Hardware einfach dazu. Die Ascom-Schutz- und Staubhüllen schützen die verschiedenen Geräte besonders wirkungsvoll durch den maßgerechten Zuschnitt. Sie gibt es für den Commodore 64, das Commodore-Disketten-Laufwerk 1541, den Commodore Drucker sowie für das IBM-PC Keyboard und die Epson Drucker MX 70/80, IDS 480 und NE 8023.

Impressum: Computronic erscheint monatlich im Verlag: Tronic-Verlagsgesellschaft mbH, Landstr. 29, 3444 Wehretal 1, Tel.: (0 56 51) 4 06 43 / 4 06 93. Redaktion: (Verantw.) Rainer Beck, Axel Gredè, Frank Brall. Freie Mitarbeiter: G. Wagner, Volker Bekker, Rolf Freitag. Gesamtherstellung: D + V Paul Dierichs GmbH + Co. KG. Kassel. Vertrieb: Inland (Groß-, Einzel- und Bahnhofsbuchhandel), sowie Österreich und Schweiz: Verlagsunion, 6200 Wiesbaden. Anzeigenverwaltung: E. Herwig. Anzeigenpreis: Es gilt die Anzeigenliste Nr. 1. Bitte Media-Unterlagen anfordern. Bezugspreise: Einzelheft (Inland) DM 5,50, Abonnement (Inland) DM 55,-, (Ausland) DM 75,-. Autoren und Manuskripte: Bei Zusendung von Manuskripten und Datenträger erteilt der Autor dem Verlag die Genehmigung zum Abdruck und Versand der veröffentlichten Programme auf Datenträger. Für die mit Namen des Verfassers gekennzeichneten Beiträge übernimmt die Redaktion lediglich die presserechtliche Verantwortung.

Computer-Börse

Verkäufe

TI 99/4A, 30 Programme, z. B. Cowboy geg. 20,– DM bei M. Morgen, Lindenstr. 66, 7333 Ebersbach

Anti-Brumm-Heiz-Adapter im formschönen Slim-Line-Gehäuse für den geplagten Spectrumfreak: 20 DM (Scheck o. Schein). Info gegen 80 Pf Rückporto. T. Schröer, Hohlstr. 11, 6791 Steinbach

Verk. **ZX 81/16 KCMEMO** + 80 Pr. + Reset + Shift Look + Rem-Ansteuerung + gr. Tastatur + Bücher + 90 Listings + Ein/Aus + gr. Netzteil für 350,- DM: Suche Spektrum 48 K, T. Bachmann, Johannisbruch 9, 2357 Springe 1

TI 99/4A + Exbasic + div. Bücher + Rec. + Kabel + Basic-Kurs, VB 750,- DM.
Tel.: 0 28 61 / 48 45
Sa. ab 18.00 Uhr

Atari/Spectrum-User-Club, monatl. Info, viele Superprogramme, Beitrag nur 5,– DM. Melden bei: ASUC, Gütersloher Str. 68, 4803 Steinhagen

C-64-Superprogramme gegen Unkosten oder Tausch. Gratisliste bei Postlagernd 058542 B, 7900 Ulm/Donau, Diskette oder Kassette

7 x 81 mit 16 K/Byte. Erweiterung + große Tastatur + ca. 30 PRG. Preis ab 230,– DM. Tel. 0 71 43 / 3 15 77

100 neue Programme für MZ 80 K, MZ 80 A, MZ 700, VZ 200, Laser 210, TI 99. Info, 1,- DM in Marken, von H. Weisel, Königsberger Straße 20, 5412 Ransbach

Verkaufe
Anrufbeantworter
Drahtlostelefon
Telefon-Anlage
Eurosignal 1200,– DM
div. USA-Telefone
Info bei:
Reinhard Winner
Höchbergerstraße 62
87 Würzburg

Software für Commodore, Tel. 02 74 72 99

Alte Hasen haben sie schon! Wann bestellen Sie Info für beste Software für Ihr Syst. Rückumschlag einsenden an M. Kiwitt, 2000 Hamburg 74, postlagernd.

Verkaufe **Taschenrechner TI 59,** Disketten 8 Zoll SS/SD, Preise VHS. Willi Hirschlein, H. v. Hohenlohestr. 44, 6990 Bad Mergentheim, Tel. 0 79 31 / 4 16 05

Verkaufe: TI-99/4A + Ext. Basic + Joysticks + Recorder + Kabel + TI-Invader + dt. Handbuch + ca. 70 PGM. + 1 TI-Magazin + Chip-Buch. VB: 700,– DM. Tel.: 02 21 / 60 76 76

Achtung: TI99/4A + Mini Memory, Editor/Assembler/Disassembler. Paket nur 30,– DM in Scheinen an T. Mielke, Falkenkamp 17b, 2000 Norderstedt 1

Here is Wimpy!
Hello User, biete für C-64 Disk beidseitig mit Topprogrammen für DM 50,- auch Tausch.
Mr. Wimpy
4350 Recklinghausen
Postfach 1017 21

Verk. TI-99/4A + Ex-Basic + Sprach-Synthesizer + Joyst.-Adapter + ca. 100 Prg. + Zubehör. VB. A. Neuburger, Feldstr. 16, 7505 Ettl. 8, Tel. 0 72 43 / 2 91 75.

TI 99/4 A Ext.-Spiel nur für Erwachsene – bitte Alter ang. zum absoluten Preishit von nur DM 10,- im Umschl. an T. Karbach, Remscheider Str. 18, 5650 Solingen 1

Wenn ihr „Mugsy von Melbourne House" wollt, dann kriegt ihr es für 35 DM (vielleicht) noch bei: Jan Weigner, Carl-Schurz-Str. 11, 2800 Bremen 1, Tel. 04 21 / 34 63 87

TI 99/4A. Erstelle Druckerlisting v. Ihren Programmen in TI- u. ext. Basic, je Pgm.-Zeile 4 Pf. + 2,50 DM f. Portokass. m. V.-Scheck o. Briefmarken an H. Schwab, Loch 21, 8504 Stein

Ti-99/4a und alle Atari-Computer für Ti:
Adventure Modul mit Cassette Mission Impossible und Pirate Adventure: 160,- DM für Atari Star Raiders für 70,- DM bei
O. Toffolo
Tel. 0 46 62 / 17 33

★★★ **FLOPPY 1541 6 x SCHNELLER !** ★★★ durch Parallelbus für C 64. Interface kpl. geprüft mit Software nur DM 179,–. Info gegen DM 1,50 in Briefmark., B. Akeson, Pf. 802, 4040 Neuss.

Ankäufe

TI-99/4A
Suche **Extended Basic Modul** evtl. Tausch gegen **Editoriassembler** sowie Maschinensprachprogramme z. B. **Flugsimulator** und **Ex. Basic Spiel Miner 2049** (USA-Version).
Tel. 0 71 32 / 3 76 08

Apple II suche Programme, Schachspiele, Flug-Simulation, versch. Vorspanntitel, Grafik für Video-Filme, T. Pavone, Heilbronnstr. 6, 7146 Tamm

Suche **Seikosha 6P-100A** Drucker mit Interface für ZX Spectrum, Lothar Schröder, Dellchenstraße 3, 6731 Frankeneck

Suche: Gute und billige Software auf Diskette für 64K Atari 600XL. Angebote an: Michael Ebeling, Riepener Straße 7, 3061 Beckedorf

Suche **Atari Diskettenstation,** Aussehen egal, Hauptsache sie funktioniert. Bitte anrufen unter Berlin 0 30 / 4 93 – 19 27

ZX Spectrum Soft- und Hardware. Preiswert durch Direktimport. Info bei: Ursula Kunz, Junge Hälden 3, 7500 Karlsruhe 41

Suche Software für: VC-LO, C-64, ZX 81, ZX Spectrum. T. Scholz, Marienstr. 13, 7778 Markdorf. Listings gesucht!

Kontakte

Hallo Spektrumfreaks. Ich möchte mit euch gerne Prog. tauschen. Liste, ob groß oder klein, an: Robert Weidinger, Schönmetzlerstr. 10, 8050 Freising oder Sams.– Diens. 0 81 61 / 43 72

Kontakt mit MZ700-Leuten gesucht. Tel. 05 11 / 42 29 75

***** Atari-Spectrum-User-Club *****
monatl. Info, bundesweit organisiert, riesige Prog.-Bibliothek usw., melden bei:
ASUC, Am Schürhof 17
4905 Spenge
Tel. 0 52 25 / 20 68

Comp.-Club
Bad Hersfeld
sucht Mitglieder!
Kontakt über
Roland Reyer
Am Giegenberg 21
6438 Ludwigsau 1
oder Jugendhaus Hersfeld

Tausch

TI-99/4A Super Ext. Basic und Assembler-progr. Tausch bzw. Verkauf Info (1,–): Stefan Schäfer, Max-Planck-Str. 8, 4050 M.-Gladbach 2. PS: Tauschliste beilegen (nur Assembl.).

C-64-Superprogramme gegen Unkosten oder Tausch. Gratisliste, Postlagernd 05 85 42 B, 7900 Ulm/Do. Nur neueste Prg.

Suche Software aller Art (Disk.), für Atari 800 XL u. Antic-Hefte bis März 1984, evtl. auch Computer u. Analog. Dietmar Levenhagen, Landwehr 83, 3057 Neustadt 1, Tel. 0 50 32 / 6 11 81

ZX-Spectrum
Software und Interessenaustausch
Monika Baumgartner
4021 Linz
Postfach 142 Austria
Bitte Liste beilegen

Computer-Börse

Verschiedenes

16 K Modul f. ZX 81, 65,- DM. Trak-Ball für Atari + Commodore, 1 Fevertaste, 99,- DM. Beides neu, unbenutzt. Verkaufe (2 DM – 4 DM) + tausche Spectr. Softw. Liste gr. T. Schröer, Hohlstr. 11, 6791 Steinbach

UFO-Forschung. Informationen über UFOs etc. kostenlos. Suche Interessenten zum Aufbau einer Computer-AG (c 64), spez. für wiss. UFO-Forschung! H. W. Peiniger, PF 2361, 5880 Lüdenscheid

VZ 200 Softw. Info g. Freiumschlag. W. Fischer, Friedrichstr. 14, 6463 Freigericht 5

Orig. engl. Software von EMJAY, (Wargame News, The War Machine), für Commodore-Rechner ab 16k und CBM 64! Näheres gegen Freiumschlag von: H. Topf, A.-Bucherer-Straße 63, 5300 Bonn 1

Die richtige Soft-Hardware? Ich sage Ihnen in welcher Zeitschrift. Info gegen frank. Rückumschlag bei: M. Kiwitt, 2000 Hamburg 74, Postlage RND

Superatari-Adventures zu Superpreisen (in Deutsch, teils mit Supergrafik). Tel. 0 77 41 / 55 56

C-64-Synthesizer: ca. 40 K, echte Noten, Partiturediting, Notenausdruck, Direkteinspielung + Notenumsetzung, Supersound. Info von Autor W. Kracht, Espellohweg 38, 2000 Hamburg 52

Roulettesoftware f. Sharp PC 1211/12/4551/PC 1500. Roulettesystemprogramme »Auf Dauer gew.« »AD Astra« »Number one« «Acclaim« »Plein Prive« »Accord« DM 550,- Cass. B. Nowotny, Giselastr. 3, 8000 München 40

An alle SF- und Fantasy-Freunde! Jeweils 8 Mitspieler für Strategiespiele gesucht. Mit Gewinnchancen. Näheres gegen Freiumschlag von: H. Topf, A.-Bucherer-Straße 63, 5300 Bonn 1

ZX-SPECTRUM! Spielend BASIC lernen Ca. mit 10 Pr.+Listing+Erklärung+Tips+Tricks für den ZX. 10,- DM bei A. Meuser, Goethestr. 18, 5205 St. Augustin NEU - SUPER ZX Spiele -

Spectrum Softw. z. B. M. Miner 8,-; Z00M 8,-; Liste + kostl. Programm anfordern, Karte genügt: H. Novak, Egerländer Pl. 19, 6053 Obertshausen 2

TI-99/4A: ca. 90 TI-Basic und ca. 80 Ex-Basic Programme zu verk. Info g. Rückumschlag. B. Knedel, Tulpengasse 16, 3171 Weyhausen, Tel. 0 53 62 / 7 11 87

Aktion: Billige Kleinanzeigen

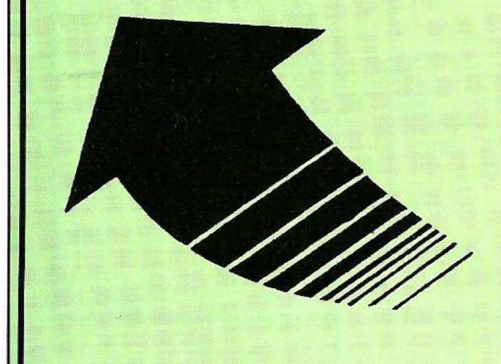
Liebe Leser,
wollen sie ein Kleinanzeige aufgeben? Suchen Sie einen gebrauchten Home-Computer, Software oder Kontakte zu Computer-Clubs etc.?

Dann nutzen Sie jetzt den Kleinanzeigenservice von „Computronic". „Computronic" ist die Zeitschrift für Home-Computer-Besitzer. Sie erreichen durch uns eine Vielzahl von Lesern im In- und Ausland.

Und so wird's gemacht:
Kleinanzeigen-Coupon ausfüllen (alle Zeilen können ausgefüllt werden) und gegen
Vorkasse von DM 4,50 einsenden an:

Tronic-Verlag
Postfach 41
3444 Wehretal 1

Also, schnell ausfüllen und einwerfen in den nächsten Postkasten. Dann ist mit Sicherheit Ihre Kleinanzeige dabei.



Achtung!

Korrektur Heft 9:

Im Programm **„Projekt"** für den C 64 wurde versehentlich die Zeile „5190 Goto 2000" (nach Zeile 5180) nicht abgedruckt.

Wir bitten um Ihr Verständnis. Die Redaktion

Computronic Kleinanzeigen nur gegen Vorkasse

Bitte veröffentlichen Sie ab nächstmöglicher Ausgabe unter der Rubrik:

☐ Verkäufe ☐ Kontakte ☐ Geschäftsempfehlungen
☐ Ankäufe ☐ Verschiedenes ☐ Tausch

folgenden Text:

Bitte jeweils 28 Buchstaben pro Zeile! Bitte Absender nicht vergessen!

Preis für Kleinanzeigen: Private Anzeige DM 4,50 je Zeile
gewerbl. Anzeige DM 8,00 je Zeile
Chiffregebühr DM 10,00 je Anzeige

Name/Vorname:

Straße, Nr.: PLZ/Ort:

Datum, Unterschrift privat ☐ gewerbl. ☐ (bitte ankreuzen)

Kassettenservice

HEFT 4

VC-64 K = 8,--DM
D = 15,--DM
Mauern, Widerstand

ZX-Spectrum K = 12,--DM
Inventur

TI 99 K = 8,50DM
Drei-Kronen-Spiel
Zahlenputzen

VC-20 K = 11,50DM
D = 18,--DM
Hangman, Saurer Regen, Quadr. Gleichung

ZX-81 K = 8,50DM
Invader, Gun-Man

Dragon 32 K = 8,--DM
Blizzard

Apple II K = 14,50DM
D = 19,50DM
Wilder Westen, Karambolage, Maskengenerator

Atari K = 10,50DM
Mastermind, Schlangenkrieg

HEFT 5

TI 99 K = 14,50DM
Karl der Käfer
Alien-Landing

VC-64 K = 15,50DM
D = 23,50DM
Space-Comets/Erdspalte/ Sprite-Data

Apple II K = 9,50DM
D = 19,50DM
Musik-Maker/Mission-Adler/Disk-Catalog

Sharp MZ 700 K = 8,50DM
Kalender
Sharp PC 1500 Lotto

Dragon 32 K = 10,--DM
Space-Flight, Geosoft

ZX-81 K = 10,--DM
Go-Ball, Grand-Prix

ZX Spectrum K = 8,50DM
Missile-Comment

Atari K = 12,50DM
Tank-Battle/Oil Panic

HEFT 6

VC-64 K = 16,50DM
D = 23,50DM
Autostart/Bestellschein/ Roadpainter

Dragon 32/64 K = 8,50DM
Wargames

Apple II K = 12,50DM
D = 19,50DM
Snake/Super Datei/Shape-tables

VC-20 K = 8,50DM
D = 15,00DM
Bestellschein/Glücky

ZX-81 K = 10,--DM
Moon-Crash/ZX-Draw

ZX-Spectrum K = 13,50DM
Defender/Lui der Wurm/ Alternativer Zeichensatz

TI-99 K = 14,50DM
Jack the Digger/Noah – 2099

HEFT 7

VC-64 K = 15,50DM
D = 19,50DM
Hardcopy/Space-Fighter/ Data-Generator

ZX-81 K = 10,--DM
Tonprogramm/Aldebaran

Atari K = 12,50DM
Startup/Zeilen-Split/ Chopper-Flight

VC-20 K = 11,--DM
D = 15,50DM
Multigraph/All-Rammer

ZX-Spectrum K = 12,50DM
Matheprogramm/Bongo-Beecatcher

Apple II K = 12,50DM
D = 19,50DM
Library/Fight

Dragon 32 K = 8,50DM
Laser-Attack

TI-99 K = 14,--DM
Lift Bär
ASC II DEF TEIL 1



Computronic **Bestellkarte-Kassettenservice**

Alle im Heft abgedruckten Programme können als zusätzlicher Service über den Verlag bezogen werden. **(Ausland nur gegen Vorkasse)**

Die Zustellung erfolgt: gegen **Vorkasse** ☐ oder Inland per **Nachnahme** ☐ + Versandkosten

innerhalb von 1 Woche

Entnehmen Sie bitte aus unserer Preisliste die notwendigen Angaben für Ihre Bestellung:

Bitte liefern Sie mir:

☐ Cassette für ________ (System, aus Heft) ☐ Anzahl

☐ Diskette für ________ (System, aus Heft) ☐ Anzahl

zum Preis von gesamt ________ DM

Name/Vorname: ________ Straße, Nr.: ________

PLZ/Ort: ________ Datum, Unterschrift ________

Kassettenservice

Jedes Programm in Computronic wird einer Endkontrolle in unserem Hause unterzogen und genauestens geprüft. Alle im Heft abgedruckten Programme sind nach der richtigen Eingabe der Listings auch funktionsfähig. Viele Leser verlieren jedoch schnell die Geduld am Programmieren, sollte etwas einmal nicht klappen. Die häufigste Ursache von Störungen im Programm, sind unterlaufene Fehler bei der Eingabe. Verzweifeln Sie nicht, sollten Sie einmal keine Zeit zum Programmieren haben oder sollte etwas nicht gelingen. Alle Programme werden im Verlag gespeichert und können jederzeit mit beiliegender Bestellkarte bezogen werden.

Tragen Sie bitte alle notwendigen Angaben in die Bestellkarte ein.

Rückgabe-Garantie:
Wir garantieren:

- kostenlosen Umtausch von defekten bzw. transportgeschädigten Datenträgern!
- die Zusendung der Umtauschware erfolgt **noch am Posteingangstag!**

Bestellung per Telefon:
Wenn es schnell gehen soll ... rufen Sie uns an. Wir nehmen Ihre Bestellung gern entgegen.

Tel.-Nr.: 0 56 51 – 4 06 93 oder **0 56 51 – 4 06 43**
Tronic-Verlag, Postfach 41, 3444 Wehretal 1
nach 17 Uhr:
Anrufbeantworter 0 56 51-4 06 93

Bitte beachten Sie:
Sie ersparen sich zusätzliche Kosten (bis zu DM 5,–), wenn Sie per Vorkasse (bar, Verrechnungsscheck) bestellen.
Ausland: ● Bestellung nur gegen Vorkasse!

Tontaubenschießen
C-64 D = 19,50 DM
K = 16,00 DM

Ein Spiel mit sehr guter Grafik für Ihren C-64. Nach den Regeln des sportlichen Tontaubenschießens müssen Sie versuchen, alle 25 Tauben, die einzeln oder auch als Dublette vom kleinen bzw. großen Haus kommen, zu treffen.

HEFT 8

System	Preis	Programme
C-64	K = 16,50DM D = 23,50DM	Monster Attack/Block-Painter/Epson-Drucker
Atari	K = 14,--DM D = 19,50DM	Painter/Hardcopy
Apple II	K = 14,--DM D = 19,50DM	Reversal Disk-Menue-Generator
TI-99	K = 14,50DM D = 19,50DM	Maya/ASC II DEF Teil 2
ASC II DEF Teil 1+2	D = 19,50DM D = 21,--DM	
Laser 2001	K = 8,50DM	Andromeda
ZX-81	K = 10,--DM	Irrgarten 3 D
ZX-Spectrum	K = 14,50DM	Solitaire/Superstat. Kleinstes gem. Vielfaches
VC-20	K = 11,--DM D = 19,50DM	Zyklo/Meteorit
Dragon	K = 13,--DM	Hardcopy

HEFT 9

System	Preis	Programme
Colour Genie	K = 10,--DM	Fuchs und Hund
C-64	K = 16,--DM D = 23,50DM	Projekt Datenbank
Atari	K = 14,--DM D = 19,50DM	The Big Quest Fünf gewinnt
Apple	D = 19,50DM	Diamonds Hilfsprogramm
TI 99	K = 14,50DM D = 19,50DM	Transfer Silverspar
Laser 2001	K = 12,--DM	Crazy Cake
ZX-81	K = 10,--DM	Reversi
ZX-Spectrum	K = 14,50DM	Jump about
VC-20	K = 14,--DM D = 19,50DM	Garten Schloß Gruselstein
Dragon	K = 10,--DM	Anwenderprogramm

HEFT 10

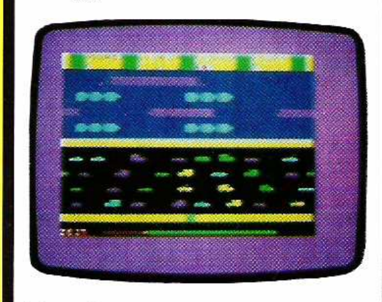
System	Preis	Programme
Ti-99	K = 14,50 D = 19,50	Mother-Duck Screen-Designer
VC-64	K = 16,50 D = 23,50	Spiders The Basic
Atari	K = 11,-- D = 18,50	Splitt
ZX-Spectrum	K = 14,50	Pac-Man Oil-Panic
VC-20	K = 14,-- D = 19,50	Fressmann Outlaw
Apple	D = 19,50	Tic – Tac – Toe Jumper
ZX-81	K = 10,--	Panik Labyrinth
Dragon 32/64	K = 14,--	Dragon paint
Laser 2001	K = 8,50	Cave-Man

Im nächsten Heft:

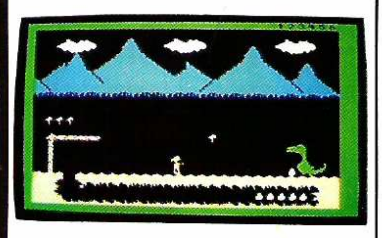
Aktuelle Informationen, Tips & Tricks Software-Reviews, Forth-Kurs und natürlich wieder tolle **Spielprogramme** und **Anwenderprogramme.** Außerdem lassen Sie sich überraschen vom **Topprogramm** der Ausgabe November.

Für den **Commodore 64** bringen wir:
Sprite-Character-Designer. – Dieses Programm übernimmt eine Vielzahl von Aufgaben. Es gestattet die Definition von Multicolor sowie normale Sprites.

Für den ZX-Spectrum: Frogger



Für den TI-99: Cave-Man

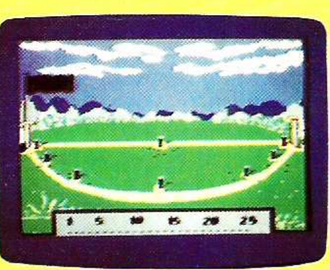


Außerdem berücksichtigen wir wieder alle bekannten Homecomputer-Systeme wie: **Commodore 64, TI-99, Atari, ZX-Spectrum, ZX-81, VC-20, Dragon 32** und den **Apple.**

Also, auf zum Zeitschriftenhändler, ab **29. 10. 1984** gibt es die neue Novemberausgabe von **„Computronic“.**

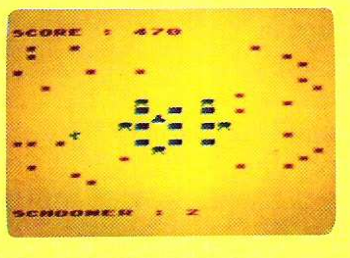
Kassettenservice aus unserem Angebot

Tontaubenschießen:



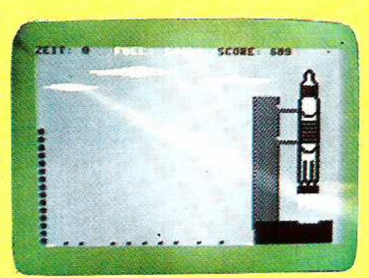
Für **Commodore 64.** Dem realistischen Tontaubenschießen nachempfunden. Von einem Katapult geschleuderte Tontauben müssen reaktionsschnell getroffen werden. In diesem Spiel kann jeder seine eigene Meisterschaft austragen. Eine Supergrafik zeichnet dieses „Tontaubenschießen" besonders aus. In Computronic wurde dieses Programm noch nicht ausgedruckt.

Tomstone-City:



Eine kleine Stadt im „Wilden Westen" gewährt Dir Schutz vor Deinen Feinden. Solange Du sie nicht verläßt kann Dir nichts passieren. Außerhalb der Stadt wirst Du jedoch gejagt. Gewählt werden kann zwischen verschiedenen Schwierigkeitsgraden. Ein Spiel für **Atari 600/800 XL.**

Projekt:



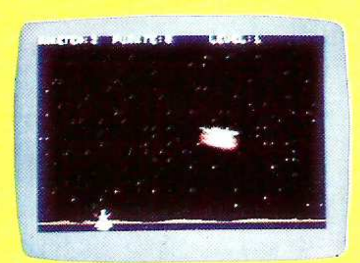
Als Topprogramm auserwählt von der Redaktion. Sie sollen eine gefährliche Mission ausführen. Um für die nächste Zeit genügend Uran zur Verfügung zu haben, sollen Sie eine Reise zum Mond unternehmen und nach dem edlen Metall suchen. Ihre Reise ist in fünf Phasen aufgegliedert, die nacheinander bewältigt werden müssen. Ein schönes Spiel für den **Commodore 64.**

Maya:



Das Spiel läuft auf dem **TI-99/4 A** mit dem **Extented Basic Modul** und wird mittels Joystick gesteuert. Der Rechner kann in diesem Spiel bis zu 378!! verschiedene Bildschirmszenen darstellen. Das Spiel verfügt über eine hervorragende grafische Darstellung. Begleiten Sie also Coconut Joe auf die abenteuerliche Expedition in den Urwald. Vielleicht haben Sie Glück und finden die Schätze der Mayas.

Monster-Attak:



Für den **Commodore 64.** Fremde Wesen greifen die Erde an. Versuchen Sie die angreifenden Monster vor Erreichen der Erde zu zerstören. Das Spiel unterteilt sich in 6 Level, wobei bei jedem Level die Spielgeschwindigkeit sowie die erreichbare Punktzahl ansteigt. Das Spiel wird mit Joystick gespielt.

Crazy-Cake:



Backe, backe Kuchen ... Setzen Sie Ihre Bäkkermütze auf, denn hier kommt ... die verrückte Kuchenbäckerei.
Das Spiel ist in Laserbais für den **Laser 2001** geschrieben und wird mit Joystick gesteuert.

Jump about:



Ein kleiner Floh und sein Freund haben ein schweres Schicksal zu meistern. Sie sind beim Herumhüpfen unbewußt in den Einflußbereich eines bösen Magiers geraten. Nun gilt es, den Weg in die Freiheit zu finden. Ein Spiel für den **ZX-Spectrum** mit toller Graphik.

Computronic erscheint monatlich im Tronic-Verlag. Der Bezugspreis beträgt im Inland DM 5,50.
Im Preis ist die gesetzliche Mehrwertsteuer enthalten.

Tronic-Verlag GmbH · Postfach 41 · 3444 Wehretal 1 · Telefon 0 56 51 - 4 06 43 / 4 06 93